AI-KU: Using Substitute Vectors and Co-Occurrence Modeling for Word Sense Induction and Disambiguation

Osman Başkaya **Enis Sert** **Volkan Cirik** **Deniz Yuret**

Artificial Intelligence Laboratory
Koç University, İstanbul, Turkey
{obaskaya,esert,vcirik,dyuret}@ku.edu.tr

Abstract

Word sense induction aims to discover different senses of a word from a corpus by using unsupervised learning approaches. Once a sense inventory is obtained for an ambiguous word, word sense discrimination approaches choose the best-fitting single sense for a given context from the induced sense inventory. However, there may not be a clear distinction between one sense and another, although for a context, more than one induced sense can be suitable. Graded word sense method allows for labeling a word in more than one sense. In contrast to the most common approach which is to apply clustering or graph partitioning on a representation of first or second order co-occurrences of a word, we propose a system that creates a substitute vector for each target word from the most likely substitutes suggested by a statistical language model. Word samples are then taken according to probabilities of these substitutes and the results of the co-occurrence model are clustered. This approach outperforms the other systems on graded word sense induction task in SemEval-2013.

1 Introduction

There exists several drawbacks of representing the word senses with a fixed-list of definitions of a manually constructed lexical database. There is no guarantee that they reflect the exact meaning of a target word in a given context since they usually contain definitions that are too general (Véronis, 2004). More so, lexical databases often include many rare senses while missing corpus/domain-specific senses (Pantel and Lin, 2004). The goal of Word Sense Induction (WSI) is to solve these problems by automatically discovering the meanings of a target word from a text, not pre-defined sense inventories. Word Sense Discrimination (WSD) approaches determine best-fitting sense among the meanings that are discovered for an ambiguous word. However, (Erk et al., 2009) suggested that annotators often gave high ratings to more than one WordNet sense for the same occurrence. They introduced a novel annotation paradigm allowing that words have more than one sense with a degree of applicability.

Unlike previous SemEval tasks in which systems labeled a target word's meaning with only one sense, word sense induction task in SemEval-2013 relaxes this by allowing a target word to have more than one sense if applicable.

Word sense induction approaches can be categorized into graph based models, bayesian, and vector-space ones. In graph-based approaches, every context word is represented as a vertex and if two context words co-occur in one or more instances of a target word, then two vertices are connected with an edge. When the graph is obtained, one of the graph clustering algorithm is employed. As a result, different partitions indicate the different senses of a target word (Véronis, 2004). Agirre et al. (2006) explored the use of two graph algorithms for unsupervised induction and tagging of nominal word senses based on corpora. Recently, Korkontzelos and Manandhar (2010) proposed a graph-based model which achieved good results on word sense induction and discrimination task in SemEval-2010.

*Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 300–306, Atlanta, Georgia, June 14-15, 2013.

Brody and Lapata (2009) proposed a Bayesian approach modeling the contexts of the ambiguous word as samples from a multinomial distribution over senses which are in turn characterized as distributions over words.

Vector-space models, on the other hand, typically create context vector by using first or second order co-occurrences. Once context vector has been constructed, different clustering algorithms may be applied. However, representing the context with first or second order co-occurrences can be difficult since there are plenty of parameters to be considered such as the order of occurrence, context window size, statistical significance of words in the context window and so on. Instead of dealing with these, we suggest representing the context with the most likely substitutes determined by a statistical language model. Statistical language models based on large corpora has been examined in (Yuret, 2007; Hawker, 2007; Yuret and Yatbaz, 2010) for unsupervised word sense disambiguation and lexical substitution. Moreover, the best results in unsupervised part-of-speech induction achieved by using substitute vectors (Yatbaz et al., 2012).

In this paper, we propose a system that represents the context of each target word by using high probability substitutes according to a statistical language model. These substitute words and their probabilities are used to create word pairs (instance id - substitute word) to feed our co-occurrence model. The output of the co-occurrence model is clustered by k-means algorithm. Our systems perform well among other submitted systems in SemEval-2013.

Rest of the paper is organized as follows. Section 2 describes the provided datasets and evaluation measures of the task. Section 3 gives details of our algorithm and is divided into five contiguous subsections that correspond to each step of our system. In Section 4 we present the differences between our three systems and their performances. Finally, Section 5 summarizes our work in this task. The code to replicate this work is available at `http://goo.gl/jPTZQ`.

2 Data and Evaluation Methodology

The test data for the graded word sense induction task in SemEval-2013 includes 50 terms containing 20 verbs, 20 nouns and 10 adjectives. There are a total of 4664 test instances provided. All evaluation was performed on test instances only. In addition, the organizers provided sense labeled trial data which can be used for tuning. This trial data is a redistribution of the Graded Sense and Usage data set provided by Katrin Erk, Diana McCarthy, and Nicholas Gaylord (Erk et al., 2009). It consists of 8 terms; 3 verbs, 3 nouns, and 2 adjectives all with moderate polysemy (4-7 senses). Each term in trial data has 50 contexts, in total 400 instances provided. Lastly, participants can use ukWaC[1], a 2-billion word web-gathered corpus, for sense induction. Furthermore, unlike in previous WSI tasks, organizers allow participants to use additional contexts not found in the ukWaC under the condition that they submit systems for both using only the ukWaC and with their augmented corpora.

The gold-standard of test data was prepared using WordNet 3.1 by 10 annotators. Since WSI systems report their annotations in a different sense inventory than WordNet 3.1, a mapping procedure should be used first. The organizers use the sense mapping procedure explained in (Jurgens, 2012). This procedure has adopted the supervised evaluation setting of past SemEval WSI Tasks, but the main difference is that the former takes into account applicability weights for each sense which is a necessary for graded word sense.

Evaluation 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. WSD evaluation is made according to three objectives:

- Their ability to detect which senses are applicable (Jaccard Index is used)
- Their ability to rank the applicable senses according to the level of applicability (Weighted Kendall's τ is used)
- Their ability to quantify the level of applicability for each sense (Weighted Normalized Discounted Cumulative Gain is used)

Clustering comparison is made by using:

[1] Available here: http://wacky.sslmit.unibo.it

- Fuzzy Normalized Mutual Information: It captures the alignment of the two clusterings independent of the cluster sizes and therefore serves as an effective measure of the ability of an approach to accurately model rare senses.

- Fuzzy B-Cubed: It 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 (i.e., sense) distribution would be equivalent.

More details can be found on the task website.[2]

3 Algorithm

In this section, we explain our algorithm. First, we describe data enrichment procedure then we will answer how each instance's substitute vector was constructed. In contrast to common practice which is clustering the context directly, we first performed word sampling on the substitute vectors and created instance id - substitute word pairs as explained in Subsection 3.3. These pairs were used in the co-occurrence modeling step described in Subsection 3.4. Finally, we clustered these co-occurrence modeling output with the k-means clustering algorithm. It is worth noting that this pipeline is performed on each target word separately.

SRILM (Stolcke, 2002) is employed on entire ukWaC corpus for the 4-gram language model to conduct all experiments.

3.1 Data Enrichment

Data enrichment aims to increase the number of instances of target words. Our preliminary experiments on the trial data showed that additional contexts increase the performance of our systems.

Assuming that our target word is *book* in noun form. We randomly fetch 20,000 additional contexts from ukWaC where our target word occurs with the same part-of-speech tag. This implies that we skip those sentences in which the word *book* functions as a verb. These additional contexts are labeled with unique numbers so that we can distinguish actual instances in the test data. We follow this procedure for every target word in the test data. In total, 1 million additional instances were fetched from ukWac. Hereafter we refer to this new dataset with as an expanded dataset.

[2] www.cs.york.ac.uk/semeval-2013/task13/

Substitute	Probability
solve	0.305
complete	0.236
meet	0.096
overcome	0.026
counter	0.022
tackle	0.014
address	0.012
...	...
...	...

Table 1: The most likely substitutes for **meet**

3.2 Substitute Vectors

Unlike other WSI methods which rely on the first or the second order co-occurrences (Pedersen, 2010), we represent the context of each target word instance by finding the most likely substitutes suggested by the 4-gram language model we built from ukWaC corpus. The high probability substitutes reflect both semantic and syntactic properties of the context as seen in Table 1 for the following example:

> And we need Your help to **meet** the challenge!

For every instance in our expanded dataset, we use three tokens each on the left and the right side of a target word as a context when estimating the probabilities for potential lexical substitutes. This tight window size might seem limited, however, tight context windows give better scores for semantic similarity, while larger context windows or second-order context words are better for modeling general topical relatedness (Sahlgren, 2006; Peirsman et al., 2008).

Fastsubs (Yuret, 2012) was used for this process and the top 100 most likely substitutes were used for representing each instance since the rest of the substitutes had negligible probabilities. These top 100 probabilities were normalized to add up to 1.0 giving us a final substitute vector for a particular target word's instance. Note that the substitute vector is a

Instance ID	Substitute Word
$meet_1$	complete
$meet_1$	solve
$meet_1$	solve
$meet_1$	overcome
...	...
...	...
$meet_1$	meet
$meet_1$	complete
$meet_1$	solve
$meet_1$	solve

Table 2: Substitute word sampling for instance $meet_1$

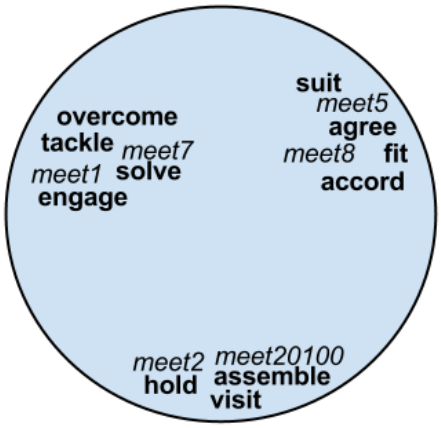

Figure 1: Co-Occcurrence Embedding Sphere for **meet**

function of the context only and is indifferent to the target word.

At the end of this step, we had 1,004,466 substitute vectors. The next common step might be to cluster these vectors either locally, which means every target word will be clustered separately; or globally, which indicates all instances (approximately 1 million) will be clustered together. Both approaches led us to lower scores than the presented method. Therefore, instead of clustering substitute vectors directly, we relied on co-occurrence modeling.

3.3 Substitute Word Sampling

Before running S-CODE (Maron et al., 2010) to model co-occurrence statistics, we needed to perform the substitute word sampling. For each target word's instance, we sample 100 substitutes from its substitute vector. Assuming that our target word is *meet* and its substitute vector is the one shown in Table 1. We choose 100 substitutes from this instance's substitute vector by using individual probabilities of substitutes. As seen in Table 2, those substitutes which have high probabilities dominate the right column. Recall that Table 2 illustrates only one instance (subscript denotes the instance number) for the target word *meet* which has 20,000 and 100 instances from the context enrichment procedure and the test, respectively. We followed the same procedure for every instance of each target word. Table 3 depicts instance id - substitute word pairs for the target word *meet* rather than for only one instance shown in Table 2.

Instance ID	Substitute Word
$meet_1$	complete
$meet_1$	solve
...	...
$meet_2$	hold
$meet_2$	visit
...	...
$meet_{20100}$	assemble
...	...
$meet_{20100}$	gather

Table 3: Substitute sampling for a target word **meet**. Instance ID - Substitute word pairs

3.4 Co-Occurrence Modeling

After sampling, we had approximately 20,000 instance id - substitute word pairs. These pairs were used to feed S-CODE. The premise is that words with similar meanings will occur in similar contexts (Harris, 1954), and at the end this procedure enables us to put together words with similar meanings as well as making the clustering procedure more accurate. If two different instances have similar substitute word pairs (i.e, similar contexts) then these two word pairs attract each other and they will be located closely on the unit sphere, otherwise they will repel and eventually be far away from each other (see Figure 1).

3.5 Clustering

We used k-means clustering on S-CODE sphere. Note that the procedures explained in the foregone subsections were repeated for each target

	System	JI	WKT	WNDCG
All Instances	ai-ku	0.759	0.804	0.432
	ai-ku(a1000)	0.759	0.794	0.612
	ai-ku(r5-a1000)	0.760	0.800	0.541
	MFS	0.381	0.655	0.337
	All-Senses	0.757	0.745	0.660
	All-Senses-freq-ranked	0.757	0.789	0.671
	All-Senses-avg-ranked	0.757	0.806	0.706
	Random-3	0.776	0.784	0.306
	Random-n	0.795	0.747	0.301

Table 4: Supervised results on the trial set using median gold-standard (JI: Jaccard Index FScore, WKT: Weighted Kendall's Tau FScore, WNDCG: Weighted Normalized Discounted Cumulative Gain FScore)

word. More precisely, the substitute sampling, co-occurrence modeling and clustering were performed one by one for each target word.

We picked 22 as k value since the test set contained words with 3 to 22 senses. After all word pairs were labeled, we counted all class labels for each instance in the test set. For example, if meet$_1$'s 50 word pairs are labeled with c_1 and 30 word pairs are labeled with c_2 and finally 20 word pairs are labeled with c_3, then this particular instance would have 50% sense$_1$, 30% sense$_2$ and 20% sense$_3$.

4 Evaluation Results

In this section, we will discuss evaluation scores and the characteristics of the test and the trial data.

All three AI-KU systems followed the same procedures described in Section 3. After clustering, some basic post-processing operations were performed for *ai-ku(a1000)* and *ai-ku(r5-a1000)*. For *ai-ku(a1000)*, we added 1000 to all sense labels which were obtained from the clustering procedure; for *ai-ku(r5-a1000)*, those sense labels occurred less than 5 times in clustering were removed since we considered them to be unreliable labels, afterwards we added 1000 for all remaining sense labels.

Supervised Metrics: Table 5 shows the performance of our systems on the test data using all instances (verbs, nouns, adjectives) for all supervised measures and in comparison with the systems that performed best and worst, most frequent sense (MFS), all senses equally weighted, all senses average weighted, random-3, and random-n baselines. Bold numbers indicate that ai-ku achieved best scores among all submitted systems. Our systems performed generally well for all three supervised measures and slightly better for all submitted systems. On the other hand, baselines achieved better scores than all participants. More precisely, on sense detection objective, MFS baseline obtained 0.552 which is the top score, while the best submitted system could reach only 0.244. Why is it the case that MFS had one of the worst sense detection score on trial data (see Table 4), but best on test data? Unlike the trial data, test data largely consists of only one sense instances, MFS usually gives correct answer. Table 6 illustrates the characteristics of the test and trial data. Instances annotated with multiple sense had a very small fraction in the test data. In fact, 517 instances in the test set were annotated with two senses (11%) and only 25 were annotated with three senses (0.5%). However, trial data provided by the organizers had almost 5 senses per instance on the average. A similar results can be observed in All-Senses baselines. On sense ranking objec-

	System	JI	WKT	WNDCG
All Instances	ai-ku	0.197	0.620	**0.387**
	ai-ku(a1000)	0.197	0.606	0.215
	ai-ku(r5-a1000)	**0.244**	**0.642**	0.332
	Submitted-Best	0.244	0.642	0.387
	All-Best	0.552	0.787	0.499
	All-Worst	0.149	0.465	0.215
	MFS	0.552	0.560	0.412
	All-Senses-eq-weighted	0.149	0.787	0.436
	All-Senses-avg-ranked	0.187	0.613	0.499
	Random-3	0.244	0.633	0.287
	Random-n	0.290	0.638	0.286

Table 5: Supervised results on the test set. (Submitted-Best indicates the best scores among all submitted system. All-Best indicates the best scores among all submitted systems and baselines. JI: Jaccard Index FScore, WKT: Weighted Kendall's Tau FScore, WNDCG: Weighted Normalized Discounted Cumulative Gain FScore)

	Trial Data	Test Data
Number of Sense	4.97	1.19
Sense Perplexity	5.79	3.78

Table 6: Average number of senses and average sense perplexity for trial and test data

	System	FScore	FNMI	FB-Cubed
All Single-sense Instances	ai-ku	**0.641**	**0.045**	0.351
	ai-ku(a1000)	0.601	0.023	0.288
	ai-ku(r5-a1000)	0.628	0.026	0.421
	Submitted-Best	0.641	0.045	0.441
	All-Best	0.641	0.048	0.570
	All-Worst	0.477	0.006	0.180
	MFS	0.578	-	-
	SemCor-MFS	0.477	-	-
	One Sense	0.569	0.0	0.570
	Random-3	0.555	0.010	0.359
	Random-n	0.533	0.006	0.223

Table 7: Supervised and unsupervised results on the test set using instances which have only one sense. Bold numbers indicate that ai-ku achieved the best submitted system scores. (FScore: Supervised FScore, FNMI: Fuzzy Normalized Mutual Information, FB-Cubed: Fuzzy B-Cubed FScore)

tives, *All-Sense-eq-weighted* outperformed all other systems. The reason is the same as the above. This baseline ranks all senses equally and since most instances had been annotated only one sense, the other wrong senses were tied and placed at the second position in ranking. As a result, this baseline achieved the highest score. Finally, for quantifying the level of applicability for each sense, Weighted NDCG was employed. *ai-ku* outperformed other submitted systems, but top score was achieved by all-sense-avg-weighted baseline. Addition to these results, organizers provided scores for instances which have only one sense. This setting contains 89% of the test data. Table 7 shows supervised and unsupervised scores for all single-sense instances. Our base system, *ai-ku*, outperformed all other system and all baselines for FScore. Moreover, it also achieved the second best score (0.045) for Fuzzy NMI. Only one baseline (*one sense per instance*) obtained slightly better score (0.048) for this metric. For Fuzzy B-Cubed, *ai-ku(r5-a1000)* obtained 0.421 which is the third best score.

Clustering Comparison: This evaluation setting 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 a WordNet sense. *ai-ku* performed best for Fuzzy NMI among other systems included baselines. For Fuzzy B-Cubed, *ai-ku(r5a1000)* outperformed random-3 and random-n baselines. Table 8 depicts the performance of our systems, best and worst systems as well as the random baselines.

	System	Fuzzy NMI	Fuzzy B-Cubed
All Instances	ai-ku	**0.065**	0.390
	ai-ku(a1000)	0.035	0.320
	ai-ku(r5-a1000)	0.039	0.451
	Submitted-Best	0.065	0.483
	All-Best	0.065	0.623
	All-Worst	0.016	0.201
	Random-2	0.028	0.474
	Random-3	0.018	0.382
	Random-n	0.016	0.245

Table 8: Scores on clustering measures (Fuzzy NMI: Fuzzy Normalized Mutual Information, Fuzzy B-Cubed: Fuzzy B-Cubed FScore)

	All instances
ai-ku	7.72
ai-ku(a1000)	7.72
ai-ku(r5-a1000)	3.11

Table 9: Average number of senses for each ai-ku systems on test data

The best scores for the graded word sense induction task in SemEval-2013 are mostly achieved by baselines in supervised setting. Major problem is that there is huge sense differences between test and trial data regarding to number of sense distribution. Participants that used trial data as for parameter tuning and picking the best algorithm achieved lower scores than baselines since test data does not show properties of trial data. Consequently, *ai-ku* systems produce significantly more senses than the gold-standard (see Table 9), and this mainly deteriorates our performance.

5 Conclusion

In this paper, we presented substitute vector representation and co-occurrence modeling on WSI task. Clustering substitute vectors directly gives lower scores. Thus, taking samples from each target's substitute vector, we obtained instance id - substitute word pairs. These pairs were used by S-CODE. Fi-

nally we run k-means on the S-CODE. Although our systems were highly ranked among the other submitted systems, no system showed better performance than the top baselines for all metrics. One explanation is that trial data does not reflect the characteristics of test data according to their number of sense distributions. Systems used trial data biased to return more than one sense for each instance since average number of sense is almost five in trial data. In addition, baselines (except random ones) know true sense distribution in the test data beforehand which make them harder to beat.

References

Eneko Agirre, David Martínez, 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 Processing,* pages 585-593.

Samuel Brody and Mirella Lapata. 2009. Bayesian Word Sense Induction. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009),* pages 103-111, Athens, Greece.

Katrin Erk, Diana McCarthy, Nicholas Gaylord. 2009. Investigations on Word Senses and Word Usages, In *Proceedings of ACL-09* Singapore.

Zellig S. Harris. 2012. Distributional structure. *Word,* Vol. 10, pages 146-162.

Tobias Hawker. 2007. USYD: WSD and lexical substitution using the Web 1T corpus In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007),* pages 207214, Prague, Czech Republic, June. Association for Computational Linguistics.

Ioannis Korkontzelos and Suresh Manandhar. 2010. UoY: Graphs of Unambiguous Vertices for Word Sense Induction and Disambiguation. In *Proceedings of the 5th International Workshop on Semantic Evaluation.* Uppsala, Sweden.

David Jurgens. 2012. An Evaluation of Graded Sense Disambiguation using Word Sense Induction. In *SemEval '12 Proceedings of the First Joint Conference on Lexical and Computational Semantics.* pages 189-198.

Yariv Maron, Michael Lamar, and Elie Bienenstock. 2012. Sphere embedding: An application to part-of-speech induction. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, editors, In *Advances in Neural Information Processing Systems 23,* pages 1567-1575.

Patrick Pantel and Dekang Lin. 2002. Discovering Word Senses from Text. In *Proceedings of the 8th ACM SIGKDD Conference,* pages 613-619, New York, NY, USA. ACM.

Ted Pedersen. 2010. Duluth-WSI: SenseClusters Applied to the Sense Induction Task of SemEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation.* pages 363-366, Uppsala, Sweden.

Yves Peirsman, Kris Heylen and Dirk Geeraerts. 2008. Size Matters. Tight and Loose Context Definitions in English Word Space Models. *In Proceedings of the ESSLLI Workshop on Distributional Lexical Semantics,* Hamburg, Germany.

Magnus Sahlgren. 2002. The Word-Space Model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces. *Ph.D. dissertation, Department of Linguistics, Stockholm University.*

Andreas Stolcke. 2002. SRILM - An Extensible Language Modeling Toolkit. *Proceedings International Conference on Spoken Language Processing,* pages 257286.

Jean Véronis. 2004. HyperLex: Lexical Cartography for Information Retrieval. *Computer Speech & Language,* 18(3):223-252.

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, EMNLP-CoNLL 2012,* July 12-14, 2012, Jeju Island, Korea.

Deniz Yuret. 2012. FASTSUBS: An Efficient Admissible Algorithm for Finding the Most Likely Lexical Substitutes Using a Statistical Language Model. *Computing Research Repository (CoRR).*

Deniz Yuret. 2007. KU: Word sense disambiguation by substitution. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007),* pages 207214, Prague, Czech Republic, June. Association for Computational Linguistics.

Deniz Yuret and Mehmet Ali Yatbaz. 2010. The noisy channel model for unsupervised word sense disambiguation. *Computational Linguistics,* Volume 36 Issue 1, March 2010, pages 111-127.