I2R: Three Systems for Word Sense Discrimination, Chinese Word Sense Disambiguation, and English Word Sense Disambiguation

Zheng-Yu Niu, Dong-Hong Ji
Institute for Infocomm Research
21 Heng Mui Keng Terrace
119613 Singapore
niu_zy@hotmail.com
dhji@i2r.a-star.edu.sg

Chew-Lim Tan
Department of Computer Science
National University of Singapore
3 Science Drive 2
117543 Singapore
tancl@comp.nus.edu.sg

Abstract

This paper describes the implementation of our three systems at SemEval-2007, for task 2 (word sense discrimination), task 5 (Chinese word sense disambiguation), and the first subtask in task 17 (English word sense disambiguation). For task 2, we applied a cluster validation method to estimate the number of senses of a target word in untagged data, and then grouped the instances of this target word into the estimated number of clusters. For both task 5 and task 17, We used the label propagation algorithm as the classifier for sense disambiguation. Our system at task 2 achieved 63.9% F-score under unsupervised evaluation, and 71.9% supervised recall with supervised evaluation. For task 5, our system obtained 71.2% micro-average precision and 74.7% macro-average precision. For the lexical sample subtask for task 17, our system achieved 86.4% coarse-grained precision and recall.

1 Introduction

SemEval-2007 launches totally 18 tasks for evaluation exercise, covering word sense disambiguation, word sense discrimination, semantic role labeling, and sense disambiguation for information retrieval, and other topics in NLP. We participated three tasks in SemEval-2007, which are task 2 (Evaluating Word Sense Induction and Discrimination Systems), task 5 (Multilingual Chinese-English Lexical Sample Task) and the first subtask at task 17 (English Lexical Sample, English Semantic Role Labeling and English All-Words Tasks).

The goal for SemEval-2007 task 2 (Evaluating Word Sense Induction and Discrimination Systems)(Agirre and Soroa, 2007) is to automatically discriminate the senses of English target words by the use of only untagged data. Here we address this word sense discrimination problem by (1) estimating the number of word senses of a target word in untagged data using a stability criterion, and then (2) grouping the instances of this target word into the estimated number of clusters according to the similarity of contexts of the instances. No sense-tagged data is used to help the clustering process.

The goal of task 5 (Chinese Word Sense Disambiguation) is to create a framework for the evaluation of word sense disambiguation in Chinese-English machine translation systems. Each participates of this task will be provided with sense tagged training data and untagged test data for 40 Chinese polysemous words. The "sense tags" for the ambiguous Chinese target words are given in the form of their English translations. Here we used a semi-supervised classification algorithm (label propagation algorithm) (Niu, et al., 2005) to address this Chinese word sense disambiguation problem.

The lexical sample subtask of task 17 (English Word Sense Disambiguation) provides sense-tagged training data and untagged test data for 35 nouns and 65 verbs. This data includes, for each target word: OntoNotes sense tags (these are groupings of WordNet senses that are more coarse-grained than tradi-

Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), pages 177–182,
Prague, June 2007. ©2007 Association for Computational Linguistics

tional WN entries), as well as the sense inventory for these lemmas. Here we used only the training data supplied in this subtask for sense disambiguation in test set. The label propagation algorithm (Niu, et al., 2005) was used to perform sense disambiguation by the use of both training data and test data.

This paper will be organized as follows. First, we will provide the feature set used for task 2, task 5 and task 17 in section 2. Secondly, we will present the word sense discrimination method used for task 2 in section 3. Then, we will give the label propagation algorithm for task 5 and task 17 in section 4. Section 5 will provide the description of data sets at task 2, task 5 and task 17. Then, we will present the experimental results of our systems at the three tasks in section 6. Finally we will give a conclusion of our work in section 7.

2 Feature Set

In task 2, task 5 and task 17, we used three types of features to capture contextual information: part-of-speech of neighboring words (no more than three-word distance) with position information, unordered single words in topical context (all the contextual sentences), and local collocations (including 11 collocations). The feature set used here is as same as the feature set used in (Lee and Ng, 2002) except that we did not use syntactic relations.

3 The Word Sense Discrimination Method for Task 2

Word sense discrimination is to automatically discriminate the senses of target words by the use of only untagged data. So we can employ clustering algorithms to address this problem. Another problem is that there is no sense inventories for target words. So the clustering algorithms should have the ability to automatically estimate the sense number of a target word.

Here we used the sequential Information Bottleneck algorithm (sIB) (Slonim, et al., 2002) to estimate cluster structure, which measures the similarity of contexts of instances of target words according to the similarity of their contextual feature conditional distribution. But sIB requires the number of clusters as input. So we used a cluster validation method to automatically estimate the sense number of a target word before clustering analysis. Cluster validation (or stability based approach)is a commonly used method to the problem of model order identification (or cluster number estimation) (Lange, et al., 2002; Levine and Domany, 2001). The assumption of this method is that if the model order is identical with the true value, then the cluster structure estimated from the data is stable against resampling, otherwise, it is more likely to be the artifact of sampled data.

Table 1: Sense number estimation procedure for word sense discrimination.

1	Set lower bound K_{min} and upper bound K_{max} for sense number k;
2	Set $k = K_{min}$;
3	Conduct the cluster validation process presented in Table 2 to evaluate the merit of k;
4	Record k and the value of M_k;
5	Set $k = k + 1$. If $k \leq K_{max}$, go to step 3, otherwise go to step 6;
6	Choose the value $\hat{k}$ that maximizes M_k, where $\hat{k}$ is the estimated sense number.

3.1 The Sense Number Estimation Procedure

Table 1 presents the sense number estimation procedure. K_{min} was set as 2, and K_{max} was set as 5 in our system. The evaluation function M_k (described in Table 2) is relevant with the sense number k. q is set as 20 here. Clustering solution which is stable against resampling will give rise to a local optimum of M_k, which indicates the true value of sense number. In the cluster validation procedure, we used the sIB algorithm to perform clustering analysis (described in section 3.2).

The function $M(C^\mu, C)$ in Table 2 is given by (Levine and Domany, 2001):

$$M(C^\mu, C) = \frac{\sum_{i,j} 1\{C^\mu_{i,j} = C_{i,j} = 1, d_i \in D^\mu, d_j \in D^\mu\}}{\sum_{i,j} 1\{C_{i,j} = 1, d_i \in D^\mu, d_j \in D^\mu\}}, \tag{1}$$

where D^μ is a subset with size $\alpha|D|$ sampled from full data set D, C and C^μ are $|D| \times |D|$ connectivity matrixes based on clustering solutions computed on D and D^μ respectively, and $0 \leq \alpha \leq 1$. The connectivity matrix C is defined as: $C_{i,j} = 1$ if d_i and d_j belong to the same cluster, otherwise $C_{i,j} = 0$. C^μ is calculated in the same way. α is set as 0.90 in this paper.

Table 2: The cluster validation method for evaluation of values of sense number k.

```
    Function: Cluster_Validation($k$, $D$, $q$)
    Input: cluster number $k$, data set $D$,
    and sampling frequency $q$;
    Output: the score of the merit of $k$;
1   Perform clustering analysis using $sIB$ on
    data set $D$ with $k$ as input;
2   Construct connectivity matrix $C_k$ based on
    above clustering solution on $D$;
3   Use a random predictor $\rho_k$ to assign
    uniformly drawn labels to instances in $D$;
4   Construct connectivity matrix $C_{\rho_k}$
    using above clustering solution on $D$;
5   For $\mu = 1$ to $q$ do
5.1   Randomly sample a subset $(D^\mu)$ with size
    $\alpha|D|$ from $D$, $0 \leq \alpha \leq 1$;
5.2   Perform clustering analysis using $sIB$ on
    $(D^\mu)$ with $k$ as input;
5.3   Construct connectivity matrix $C_k^\mu$ using
    above clustering solution on $(D^\mu)$;
5.4   Use $\rho_k$ to assign uniformly drawn labels
    to instances in $(D^\mu)$;
5.5   Construct connectivity matrix $C_{\rho_k}^\mu$
    using above clustering solution on $(D^\mu)$;
    Endfor
6   Evaluate the merit of $k$ using following
    objective function:
    $M_k = \frac{1}{q}\sum_\mu M(C_k^\mu, C_k) - \frac{1}{q}\sum_\mu M(C_{\rho_k}^\mu, C_{\rho_k})$,
    where $M(C^\mu, C)$ is given by equation (1);
7   Return $M_k$;
```

$M(C^\mu, C)$ measures the proportion of document pairs in each cluster computed on D that are also assigned into the same cluster by clustering solution on D^μ. Clearly, $0 \leq M \leq 1$. Intuitively, if cluster number k is identical with the true value, then clustering results on different subsets generated by sampling should be similar with that on full data set, which gives rise to a local optimum of $M(C^\mu, C)$.

In our algorithm, we normalize $M(C_{F,k}^\mu, C_{F,k})$ using the equation in step 6 of Table 2, which makes our objective function different from the figure of merit (equation (1)) proposed in (Levine and Domany, 2001). The reason to normalize $M(C_{F,k}^\mu, C_{F,k})$ is that $M(C_{F,k}^\mu, C_{F,k})$ tends to decrease when increasing the value of k. Therefore for avoiding the bias that smaller value of k is to be selected as cluster number, we use the cluster validity of a random predictor to normalize $M(C_{F,k}^\mu, C_{F,k})$.

3.2 The sIB Clustering Algorithm

Here we used the sIB algorithm (Slonim, et al., 2002) to estimate cluster structure, which measures the similarity of contexts of instances according to the similarity of their feature conditional distribution. sIB is a simplified "hard" variant of information bottleneck method (Tishby, et al., 1999).

Let d represent a document, and w represent a feature word, $d \in D$, $w \in F$. Given the joint distribution $p(d, w)$, the document clustering problem is formulated as looking for a compact representation T for D, which preserves as much information as possible about F. T is the document clustering solution. For solving this optimization problem, sIB algorithm was proposed in (Slonim, et al., 2002), which found a local maximum of $I(T, F)$ by: given an initial partition T, iteratively drawing a $d \in D$ out of its cluster $t(d)$, $t \in T$, and merging it into t^{new} such that $t^{new} = argmax_{t \in T}\mathbf{d}(d, t)$. $\mathbf{d}(d, t)$ is the change of $I(T, F)$ due to merging d into cluster t^{new}, which is given by

$$\mathbf{d}(d, t) = (p(d) + p(t))JS(p(w|d), p(w|t)). \quad (2)$$

$JS(p, q)$ is the Jensen-Shannon divergence, which is defined as

$$JS(p, q) = \pi_p D_{KL}(p\|\overline{p}) + \pi_q D_{KL}(q\|\overline{p}), \quad (3)$$

$$D_{KL}(p\|\overline{p}) = \sum_y plog\frac{p}{\overline{p}}, \quad (4)$$

$$D_{KL}(q\|\overline{p}) = \sum_y qlog\frac{q}{\overline{p}}, \quad (5)$$

$$\{p, q\} \equiv \{p(w|d), p(w|t)\}, \quad (6)$$

$$\{\pi_p, \pi_q\} \equiv \{\frac{p(d)}{p(d) + p(t)}, \frac{p(t)}{p(d) + p(t)}\}, \quad (7)$$

$$\overline{p} = \pi_p p(w|d) + \pi_q p(w|t). \quad (8)$$

4 The Label Propagation Algorithm for Task 5 and Task 17

In the label propagation algorithm (LP) (Zhu and Ghahramani, 2002), label information of any vertex in a graph is propagated to nearby vertices through weighted edges until a global stable stage is achieved. Larger edge weights allow labels to travel through easier. Thus the closer the examples, more likely they have similar labels (the global consistency assumption).

In label propagation process, the soft label of each initial labeled example is clamped in each iteration to replenish label sources from these labeled data. Thus the labeled data act like sources to push out labels through unlabeled data. With this push from labeled examples, the class boundaries will be pushed through edges with large weights and settle in gaps along edges with small weights. If the data structure fits the classification goal, then LP algorithm can use these unlabeled data to help learning classification plane.

Let $Y^0 \in N^{n\times c}$ represent initial soft labels attached to vertices, where $Y^0_{ij} = 1$ if y_i is s_j and 0 otherwise. Let Y^0_L be the top l rows of Y^0 and Y^0_U be the remaining u rows. Y^0_L is consistent with the labeling in labeled data, and the initialization of Y^0_U can be arbitrary.

Optimally we expect that the value of W_{ij} across different classes is as small as possible and the value of W_{ij} within same class is as large as possible. This will make label propagation to stay within same class. In later experiments, we set σ as the average distance between labeled examples from different classes.

Define $n \times n$ probability transition matrix $T_{ij} = P(j \rightarrow i) = \frac{W_{ij}}{\sum_{k=1}^{n} W_{kj}}$, where T_{ij} is the probability to jump from example x_j to example x_i.

Compute the row-normalized matrix $\overline{T}$ by $\overline{T}_{ij} = T_{ij}/\sum_{k=1}^{n} T_{ik}$. This normalization is to maintain the class probability interpretation of Y.

Then LP algorithm is defined as follows:

1. Initially set t=0, where t is iteration index;

2. Propagate the label by $Y^{t+1} = \overline{T}Y^t$;

3. Clamp labeled data by replacing the top l row of Y^{t+1} with Y^0_L. Repeat from step 2 until Y^t converges;

4. Assign $x_h (l + 1 \leq h \leq n)$ with a label $s_{\hat{j}}$, where $\hat{j} = argmax_j Y_{hj}$.

This algorithm has been shown to converge to a unique solution, which is $\widehat{Y}_U = \lim_{t\rightarrow\infty} Y^t_U = (I - \overline{T}_{uu})^{-1}\overline{T}_{ul}Y^0_L$ (Zhu and Ghahramani, 2002). We can see that this solution can be obtained without iteration and the initialization of Y^0_U is not important, since Y^0_U does not affect the estimation of $\widehat{Y}_U$. I is $u \times u$ identity matrix. $\overline{T}_{uu}$ and $\overline{T}_{ul}$ are acquired by splitting matrix $\overline{T}$ after the l-th row and the l-th column into 4 sub-matrices.

For task 5 and 17, we constructed connected graphs as follows: two instances u, v will be connected by an edge if u is among v's k nearest neighbors, or if v is among u's k nearest neighbors as measured by cosine or JS distance measure. k is set 10 in our system implementation.

5 Data Sets of Task 2, Task 5 and Task 17

The test data for task 2 includes totally 27132 untagged instances for 100 ambiguous English words. There is no training data for task 2.

There are 40 ambiguous Chinese words in task 5. The training data for this task consists of 2686 instances, while the test data includes 935 instances.

There are 100 ambiguous English words in the first subtask of task 17. The training data for this task consists of 22281 instances, while the test data includes 4851 instances.

6 Experimental Results of Our Systems at Task 2, Task 5 and Task 17

Table 3: The best/worst/average F-score of all the systems at task 2 and the F-score of our system at task 2 for all target words, nouns and verbs with unsupervised evaluation.

	All words	Nouns	Verbs
Best	78.7%	80.8%	76.3%
Worst	56.1%	65.8%	45.1%
Average	65.4%	69.0%	61.4%
Our system	63.9%	68.0%	59.3%

Table 3 lists the best/worst/average F-score of all the systems at task 2 and the F-score of our system at task 2 for all target words, nouns and verbs with

Table 4: The best/worst/average supervised recall of all the systems at task 2 and the supervised recall of our system at task 2 for all target words, nouns and verbs with supervised evaluation.

	All words	Nouns	Verbs
Best	81.6%	86.8%	75.7%
Worst	78.5%	81.4%	75.2%
Average	79.6%	83.0%	75.7%
Our system	81.6%	86.8%	75.7%

Table 5: The best/worst/average micro-average precision and macro-average precision of all the systems at task 5 and the micro-average precision and macro-average precision of our system at task 5.

	Micro-average	Macro-average
Best	71.7%	74.9%
Worst	33.7%	39.6%
Average	58.5%	62.7%
Our system	71.2%	74.7%

unsupervised evaluation. Our system obtained the fourth place among six systems with unsupervised evaluation. Table 4 shows the best/worst/average supervised recall of all the systems at task 2 and the supervised recall of our system at task 2 for all target words, nouns and verbs with supervised evaluation. Our system is ranked as the first among six systems with supervised evaluation. Table 7 lists the estimated sense numbers by our system for all the words at task 2. The average of all the estimated sense numbers is 3.1, while the average of all the ground-truth sense numbers is 3.6 if we consider the sense inventories provided in task 17 as the answer. It seems that our estimated sense numbers are close to the ground-truth ones.

Table 5 provides the best/worst/average micro-average precision and macro-average precision of all the systems at task 5 and the micro-average precision and macro-average precision of our system at task 5. Our system obtained the second place among six systems for task 5.

Table 6 shows the best/worst/average coarse-grained score (precision) of all the systems the lexical sample subtask of task 17 and the coarse-grained score (precision) of our system at the lexical sample subtask of task 17. The attempted rate of all the systems is 100%. So the precision value is equal to the recall value for all the systems. Here we listed only the precision for the 13 systems at this subtask. Our system is ranked as the third one among 13 systems.

Table 6: The best/worst/average coarse-grained score (precision) of all the systems at the lexical sample subtask of task 17 and the coarse-grained score (precision) of our system at the lexical sample subtask of task 17.

	Coarse-grained score (precision)
Best	88.7%
Worst	52.1%
Average	70.0%
Our system	86.4%

7 Conclusion

In this paper, we described the implementation of our $I2R$ systems that participated in task 2, task 5, and task 17 at SemEval-2007. Our systems achieved 63.9% F-score and 81.6% supervised recall for task 2, 71.2% micro-average precision and 74.7% macro-average precision for task 5, and 86.4% coarse-grained precision and recall for the lexical sample subtask of task 17. The performance of our system is very good under supervised evaluation. It may be explained by that our system has the ability to find some minor senses so that it can outperforms the baseline system that always uses the most frequent sense as the answer.

References

Agirre E. , & Soroa A. 2007. SemEval-2007 Task 2: Evaluating Word Sense Induction and Discrimination Systems. *Proceedings of SemEval-2007, Association for Computational Linguistics.*

Lange, T., Braun, M., Roth, V., & Buhmann, J. M. 2002. Stability-Based Model Selection. *Advances in Neural Information Processing Systems 15.*

Lee, Y.K., & Ng, H.T. 2002. An Empirical Evaluation of Knowledge Sources and Learning Algorithms for Word Sense Disambiguation. *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, (pp. 41-48).

Levine, E., & Domany, E. 2001. Resampling Method for Unsupervised Estimation of Cluster Validity. *Neural Computation*, Vol. 13, 2573–2593.

Niu, Z.Y., Ji, D.H., & Tan, C.L. 2005. Word Sense Disambiguation Using Label Propagation Based Semi-Supervised Learning. *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*.

Slonim, N., Friedman, N., & Tishby, N. 2002. Unsupervised Document Classification Using Sequential Information Maximization. *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Tishby, N., Pereira, F., & Bialek, W. (1999) The Information Bottleneck Method. *Proc. of the 37th Allerton Conference on Communication, Control and Computing*.

Zhu, X. & Ghahramani, Z.. 2002. Learning from Labeled and Unlabeled Data with Label Propagation. *CMU CALD tech report CMU-CALD-02-107*.

Table 7: The estimated sense numbers by our system for all the words at task 2.

explain	2	move	3
position	3	express	4
buy	2	begin	2
hope	3	prepare	3
feel	5	policy	2
hold	2	attempt	2
work	5	recall	3
people	4	find	2
system	2	join	2
bill	2	build	2
hour	5	base	3
value	4	management	2
job	5	turn	4
rush	2	kill	2
ask	2	area	5
approve	4	affect	4
capital	4	keep	5
purchase	2	improve	2
propose	2	do	2
see	3	drug	5
president	3	come	5
power	3	disclose	4
effect	2	avoid	3
part	5	plant	2
exchange	4	share	2
state	2	carrier	2
care	5	complete	2
promise	3	maintain	3
estimate	2	development	4
rate	2	space	5
say	2	raise	3
remove	5	future	3
grant	4	network	3
remember	3	announce	5
cause	2	start	3
point	5	order	2
occur	4	defense	5
authority	3	set	3
regard	2	chance	2
go	3	produce	2
allow	4	negotiate	2
describe	2	enjoy	4
prove	3	exist	4
claim	4	replace	3
fix	2	examine	3
end	5	lead	3
receive	3	source	2
complain	3	report	2
need	2	believe	2
condition	2	contribute	3