Semantic Relation Extraction Based on Semi-supervised Learning

Haibo Li, Yutaka Matsuo, and Mitsuru Ishizuka

University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo, Japan
lihaibo@mi.ci.i.u-tokyo.ac.jp,
matsuo@biz-model.t.u-tokyo.ac.jp,
ishizuka@i.u-tokyo.ac.jp

Abstract. Many tasks of information extraction or natural language processing have a property that the data naturally consist of several views—disjoint subsets of features. Specifically, a semantic relationship can be represented with some entity pairs or contexts surrounding the entity pairs. For example, the *Person-Birthplace* relation can be recognized from the *entity pair* view, such as (*Albert Einstein, Ulm*), (*Pablo Picasso, Malaga*) and so on. On the other side, this relation can be identified with some contexts, such as "*A was born in B*", "*B, the birth place of A*" and so on.

To leverage the unlabeled data in the training stage, semi-supervised learning has been applied to relation extraction task. In this paper, we propose a multi-view semi-supervised learning algorithm, Co-Label Propagation, to combine the 'information' from both the *entity pair* view and the *context* view. In propagation process, the label scores of classes are spread not only in the *entity pair* view and the *context* view, but also between the two views. The proposed algorithm is evaluated using semantic relation classification tasks. The experiment results validate its effectiveness.

Keywords: semi-supervised learning, multi-view learning, relation extraction.

1 Introduction

Relationship extraction is a task of recognizing a particular relationship between two or more entities in documents. However, a large amount of manually labeled data is demanded when the supervised learning methods are used to address this problem. But annotating training data is a very tedious and time consuming work [1]. Meanwhile, semi-supervised learning addresses this problem by combining a large amount of unlabeled data with a small set of labeled data to train a classifier, such as co-training, label propagation and so on.

Recently, label propagation, a graph-based semi-supervised learning method, has increasingly attracted research attention [1,2,3]. The label propagation algorithm constructs a graph with both labeled and unlabeled data. The seed nodes in the graph propagate their labels to neighbors according to their similarity. In label propagation process, the label distribution of initial labeled seeds are clamped in each iteration to replenish

P.-J. Cheng et al. (Eds.): AIRS 2010, LNCS 6458, pp. 270–279, 2010.

the label sources from these labeled data. With this spreading from labeled examples, the class boundaries are spread through edges with large weights and settle in gaps along edges with low weights.

Generally, many tasks of information extraction or natural language processing have a property that the data naturally consist of several views—disjoint subsets of features. For instance, web pages can be described by their contents or hyperlinks pointing to these pages [4]. A popular paradigm of multi-view learning is the co-training algorithm, which splits all features into two subsets and trains two classifiers by the labeled seeds in each view. Each classifier classifies the unlabeled data in the unlabeled data pool and provides the other classifier with the few unlabeled examples as training seeds that receive the highest confidence from the first classifier.

In relation classification task, a semantic relationship can be represented with two different kinds of "information": the entity pair itself and the context surrounding it. Given an instance of data for relation classification as follows:

$$s = (C_{pre}, e_1, C_{mid}, e_2, C_{post})$$

where e_1 and e_2 are nouns or noun phrases and C_{pre}, C_{mid}, and C_{post} are the contexts before, between, and after the nominal pairs. We split s into two parts: entity pair (e_1, e_2) and contexts $(C_{pre}, C_{mid}, C_{post})$. Many features can be extracted from the two parts respectively and applied to learning.

In this paper, we propose a label propagation based multi-view learning algorithm, Co-Label Propagation (Co-LP), which combines the information of the *entity pair* view and the *context* view. Let $S = \{s_i | i = 1, \cdots, u\}$ be a set of sentence tuples. A instance $s_i = (a_j, b_k)$ is composed of two parts: entity pair a_j and context b_k. Let $A = \{a_j | j = 1, \cdots, n\}$ and $B = \{b_k | k = 1, \cdots, m\}$ be sets of entity pairs and contexts respectively. A entity pair a_j occurs in S at least once with one or more context(s). A context b_k signify at least one or more entity pair(s). The proposed Co-LP algorithm constructs three graphs: $G_A =< A, E_A >$, $G_B =< B, E_B >$ and $G_{AB} =< A \cup B, E_{AB} >$. A and B represent the data point set of *entity pair* view and *context* view respectively, and E_A, E_B are edges that connect intra-view data points. Graphs G_A and G_B represent the similarities among data points in each view respectively. The inter-view graph G_{AB} is a bipartite graph which describes the correlation between data points of two different views. The graph G_{AB} uses the correlation of the entity pair a_j and context b_k to combine the label score of data points in the two views.

The remainder of this paper is organized as follows: In section 2, we outline related works of relation extraction and semi-supervised learning. In section 3, we present our Co-Label Propagation algorithm in detail. Section 4 presents some experiments and discussion of the results. Finally, in section 5, we discuss our conclusions.

2 Related Work

To leverage the unlabeled data in the training stage, semi-supervised learning has been applied to relation extraction task. As mentioned in previous section, Chen et al. explored a graph based semi-supervised learning for relation extraction, which makes use

of unlabeled data [5]. Niu et al. investigated label propagation for a word-sense disambiguation task [6].

Co-Training is a semi-supervised, multi-view algorithm that uses the initial seeds to learn a classifier in each view [4]. Then each classifier is applied to classify all unlabeled data. The examples on which each classifier makes the most confident predictions are selected and added to the training set. Based on the new training set, a new classifier is learned in each view, and the whole process is repeated for several iterations.

The proposed Co-LP is based on label propagation which models an entire dataset as a weighted graph and propagates labels through the graph along its high-density areas [3]. Zhou et al. proposed another graph-based algorithm, the local and global consistency algorithm, in which the function at each node receives contribution from its neighbors in each step [2].

3 Co-label Propagation

3.1 Preliminaries

We presume that each data point s has two views—$s =< a, b >$—where a and b denote data points constructed respectively from *entity pair* view and *context* view. Let u be the number of data points in the feature space built with all features. Similarly, n and m respectively signify the quantities of data points in the feature space generated with *entity pair* view and *context* view. More formally, the dataset $S \subseteq A \times B$, $|S| = u$, $|A| = n$, $|B| = m$, where $u \geq n, m$, each example $s \in S$ is given as (a, b). Actually, $L = \{l_1, l_2, \cdots, l_c\}$ is the set of labels and $|L| = c$.

Let $T^A = (T^A_{ij}, \quad i, j = 1, 2, \cdots, n)$ be an $n \times n$ similarity matrix constructed from A in which T^A_{ij} represents the similarity between a_i and a_j calculated from *entity pair* view. Let T^B be defined similarly, as shown above from B.

Let $W^{AB} = (W^{AB}_{ij}, \quad i = 1, 2, \cdots, n; j = 1, 2, \cdots, m)$ be an $n \times m$ matrix defined as the correlation matrix between *entity pair* view and *context* view. In addition, T^{AB}=$(T^{AB}_{ij}, \quad i = 1, 2, \cdots, n; j = 1, 2, \cdots, m)$ is the row-normalized W^{AB}, as Eq. 1 shows, where T^{AB}_{ij} denotes the normalized correlation between a_i and b_j. In addition, T^{BA}=$(T^{BA}_{ij}, \quad i = 1, 2, \cdots, m; j = 1, 2, \cdots, n)$ is the transposed matrix of column-normalized W^{AB} as presented in Eq. 2.

$$[T^{AB}]_{ij} = \frac{[W^{AB}]_{ij}}{\sum_{k=0}^{m}[W^{AB}]_{ik}}. \tag{1}$$

$$[T^{BA}]_{ij} = \frac{[W^{BA}]_{ji}}{\sum_{k=0}^{n}[W^{BA}]_{ki}}. \tag{2}$$

Let Y^A_t be a $n \times c$ labeling matrix, where $[Y^A_t]_{ij}$ denotes the probability of a_i labeled as l_j in *t-th* round propagation. Let Y^A_0 be initialized by the labeled data as

$$[Y^A_0]_{ij} = \begin{cases} 1 \text{ if } a_i \text{ is labeled as } l_j \\ 0 \text{ otherwise} \end{cases}.$$

Similarly, Y_t^B is an $m \times c$ labeling matrix, whose i-*th* row represents the label probability distribution of data point b_i and Y_0^B is initialized similarly as Y_0^A. Let Y be an $u \times c$ labeling matrix, where $[Y]_{ij}$ denotes the label probability distribution of data point $s_i \in S$.

Table 1. Co-Label Propagation algorithm

Given:
- Intra-view similarity matrices T^A, T^B
- Inter-view correlation matrices T^{AB}, T^{BA}
- Initial label matrices of each view Y_0^A, Y_0^B

1. Propagate in each view

$$[Y_{t+1}^A]' \leftarrow T^A Y_t^A$$
$$[Y_{t+1}^B]' \leftarrow T^B Y_t^B$$

2. Propagate between different views

$$[Y_{t+1}^A]'' \leftarrow T^{AB} Y_t^B$$
$$[Y_{t+1}^B]'' \leftarrow T^{BA} Y_t^A$$

3. Combine the label score

$$Y_{t+1}^A \leftarrow [Y_{t+1}^A]' + [Y_{t+1}^A]''$$
$$Y_{t+1}^B \leftarrow [Y_{t+1}^B]' + [Y_{t+1}^B]''$$

4. Row-normalize Y_{t+1}^A and Y_{t+1}^B to maintain the class propagation interpretation.

$$[Y_t^A]_{ij} = [Y_t^A]_{ij} / \sum_{k=0}^{c} [Y_t^A]_{ik}$$

$$[Y_t^B]_{ij} = [Y_t^B]_{ij} / \sum_{k=0}^{c} [Y_t^B]_{ik}$$

5. Clump the labeled data: replace the rows of labeled data in Y_t^A and Y_t^B with the corresponding rows of Y_0^A and Y_0^B respectively:

6. Repeat from step 2 for f times:

7. Combine the two matrices: Y_t^A and Y_t^B. For each node $s_i =< a_j, b_k >$, construct an $u \times c$ matrix Y with the labeling matrix of both views.

$$[Y]_{il} = [Y_t^A]_{jl} [Y_t^B]_{kl}$$

3.2 Intra-view and Inter-view Label Propagation

In the proposed algorithm, we construct two intra-view graphs: $G_A =< A, E_A >$, $G_B =< B, E_B >$; one inter-view graph $G_{AB} =< A \cup B, E_{AB} >$. A, B represents the data point in *entity pair* view and where *context* view. G_{AB} represent similarities between data points in different views. Following [1], we use both labeled and unlabeled nodes to create a fully connected intra-view graph in each view. The edge between node i, j is weighted as

$$T_{ij} = \exp\Big(- \frac{sim_{ij}^2}{\alpha^2} \Big). \tag{3}$$

where sim_{ij} signifies the similarity of x_i and x_j calculated using some similarity measure, and α is used to control the weight. As described in this paper, we set α as the average similarity between labeled examples from different classes.

In the first step of propagation, every node in each view receives a contribution from the linked nodes in the same view. The second step is to spread label scores among different views. We build an inter-view graph $G_{AB} =< A \cup B, E_{AB} >$ between *entity pair* view and *context* view, which is used to amend the class distributions of each node. The weight of edge in E_{AB} is given as W^{AB}. In addition, T^{AB} is row-normalized by Eq. 1 and T^{BA} is line-normalized by Eq. 2. Table 1 presents the proposed algorithm concretely.

3.3 Rebalance the Label Distribution Using Label Bidding

After the matrix Y is learned, the label bidding process is executed to rebalance the label distribution. When data classes are very close or when labeled data are very few, rebalancing the label distribution can improve the final classification performance. Initially, the number of each type of labels can be estimated from labeled data. In the label bidding process, the learned label score of each node in Y is regarded as a bid for the labels. For example, if Y_{il} is the highest bid currently and class l has labels remained, then data point i is labeled as l. Then the data point i exits from bidding and the label number of class l subtracts 1. The second highest bid is processed if class l has no labels. This process will be repeated until all the labels are 'sold'.

4 Experiments

4.1 Evaluation of the Semantic Relation Classification Performance

In this section, we present our empirical study using SemEval-2007 Task 04: Classification of Semantic Relations between Nominals [7]. This dataset consists of seven semantic relations and every semantic relation is a separate binary classification task. Table 2 presents the number of positive and negative examples of each relation.

In our experiment, we first put the training set and test set of SemEval-07 Task 04 together; then we randomly select different percentages of data point as labeled seeds and others as unlabeled data. All data are projected into context and nominal pair views. To test the efficiency of inter-view label propagation algorithm, we also use the other algorithm named UnIVLP. This algorithm uses the same feature splitting and only propagates label scores in each view. In other words, UnIVLP merely skip over steps 2 and 3 portrayed in Table 1.

Table 2. SemEval-07 Task 04 Dataset Statistics

Relation Type	Training Data (positive)	Test Data (positive)
Cause-Effect	140 (65)	80 (41)
Instrument-User	140 (65)	78 (38)
Product-Producer	140 (65)	93 (62)
Origin-Entity	140 (65)	81 (36)
Theme-Tool	140 (65)	71 (29)
Part-Whole	140 (65)	72 (26)
Content-Container	140 (65)	74 (38)

Intra-View Similarity. To weight the similarity graph of each view, we use a frequently used measure, the cosine similarity measure (as Eq. 4), to calculate the similarity between any two data nodes in each view.

$$\cos(x, y) = \frac{\sum_{i=1}^{n} x_i \cdot y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}. \tag{4}$$

Matrices T^A and T^B, are constructed and normalized as previous mentioned.

Inter-View Similarity. Mutual information is an efficient measure of the relation between two random variables. Therefore, we use mutual information (as Eq. 5) between a_j and b_k to measure the relevance between data points of different views. Actually, $P(a_j)$, $P(b_k)$ respectively represent the probabilities of node a_j and b_k in V. Furthermore, $P(a_j, b_k)$ is the joint probability distribution of the node pair.

$$[W^{AB}]_{jk} = \log_2 \frac{P(a_j, b_k)}{P(a_j) \cdot P(b_k)}. \tag{5}$$

Features and View Splitting. In this experiment, as Table 3 shows, we use 13 features extracted from the SemEvald-07 Task 04 dataset. Because of the property of this problem, the features are divisible into two subsets: Context and Nominal pair. We put the first four features in Table 3 into the Nominal pair feature set; all other features are distributed to the Context feature set. It is different general semantic relation classification tasks, we only used the surface token of nominals and contexts. Because we specifically examine testing the classification performance of the algorithm, the syntactic features of words are unimportant.

Experiment Results. For semantic relations of all types, the proposed method is compared with the algorithms: 1) using label propagation, treating all features as one view (LP-ALL); 2) using label propagation in each view without inter-view propagation (UnIVLP);

Figure 1 presents the average accuracy of seven relation types. It is apparent from Figure 1 that inter-view propagation can reduce the classification error in most cases.

Table 3. Features used in experiment

Feature	Description
Nom1	Surface tokens of the first nominal
Nom2	Surface tokens of the second nominal
N1_ NET	WordNet senses of the first nominal
N2_ NET	WordNet senses of the second nominal
WBNUL	Whether a word exists in between
WBO1	The only word in between when only one is word in between
WBF	The first word in between when at least two words are in between
WBL	The last word in between when at least two words are in between
WBO	Other words in between except the first and last words when at least three words are in between
BN1F	The first word before the first nominal
BN1L	The second word before the first nominal
AN2F	The first word after the second nominal
AN2L	The second word after the second nominal

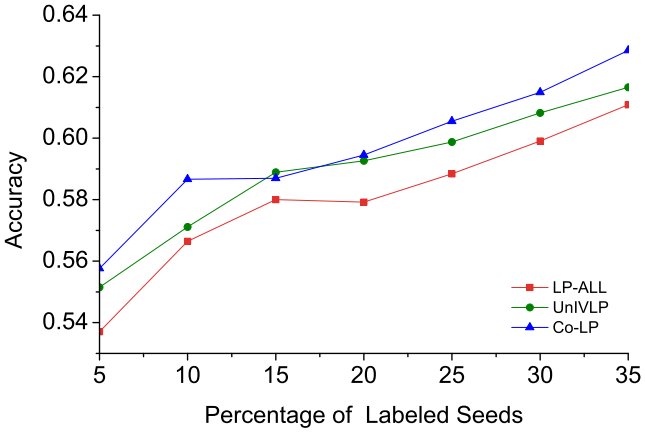

Fig. 1. Average Accuracy of seven relation types in the in SemEval-07 dataset

However, when the labeled seed percentage is 15%, Co-LP cannot beat UnIVLP, indicating that inter-view exchanging the label score cannot improve the performance of the classifier in these cases. Comparing UnIVLP with LP-ALL, it is apparent that UnIVLP works better than single view label propagation, although UnIVLP linearly combines results from separate views. Comparing Co-LP, UnIVLP with LP-ALL, the practice of regarding all features as two views always outperforms their treatment as a single view: Co-LP achieves the best accuracy. Therefore, a classifier trained on one view cannot provide useful information to another classifier. One possible explanation of Co-LP outperforming LP-ALL is that the feature space is provided with a considerable amount of redundancy. This redundancy, in effect, improved the classification accuracy.

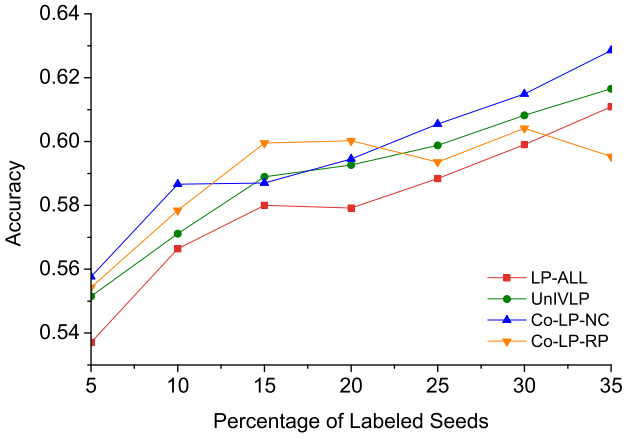

Fig. 2. Average classification accuracy of Co-LP with the Random Projected feature

Comparing Co-LP with UnIVLP, Co-LP employs the relation of different views; such information might improve the classification accuracy.

4.2 Different Feature Splitting

In this experiment, we specifically examine the sensitivity of Co-LP to feature splitting. This will elucidate the dependence of the proposed algorithm for feature splitting. To this end, we split the feature randomly into two disjoint subsets and repeat the above semantic relation classification experiment on the same dataset. We compare the new feature splitting to the (nominal pair, context) feature splitting (Co-LP-NC) in the experiment presented above. We can observe from Figure 2 that, although we randomly split the feature (Co-LP-RP), Co-LP-RP still outperforms label propagation (LP-ALL) in most cases. With labeled data of 15 and 20 percent, Co-LP-RP even outperforms Co-LP-NC, but after 25 percent, Co-LP-NC works better than either of the other two algorithms. When we randomly label 35 percentages of data as seeds, the proposed algorithm with random projected features (Co-LP-RP) cannot beat the LP-ALL algorithm.

4.3 Robustness with Respect to the Inter-view Correlation Measure

For studying the robustness of the proposed method to the correlation measures between different views, we use two similarity measures that are often used in the Natural Language Processing community: the matching coefficient and dice coefficient. The Eq. 6 and Eq. 7 respectively show the Matching coefficient and Dice coefficient. We use the same sampled data to test the sensitivity of Co-LP algorithm to the correlation measure.

– Matching coefficient:

$$[W^{AB}]_{jk} = P(a_j, b_k). \tag{6}$$

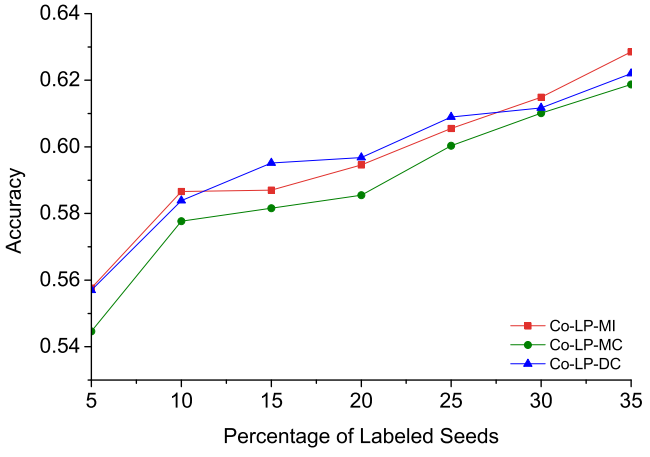

Fig. 3. Average classification accuracy of Co-LP with different correlation measures

– Dice coefficient:

$$[W^{AB}]_{jk} = \frac{2 \cdot P(a_j, b_k)}{P(a_j) + P(b_k)}. \tag{7}$$

Figure 3 portrays the average accuracies of different correlation measures: the mutual information measure (MI), matching coefficient (MC), and Dice coefficient (DC). We observe that the accuracy using DC closely resembles that using MI. Furthermore, we can find that the accuracies of MI and DC are each better than that used MC.

5 Conclusion

In this paper, we propose the Co-Label Propagation algorithm, which is based on the generalized cluster assumption. The experiment results show that our proposed algorithm can improve the performance of the label propagation algorithm, a well-known graph-based algorithm, on the SemEval-07 task 04 dataset. We also show that Co-LP works well with different splitting of feature set and choice of the inter-view correlation measure.

References

1. Zhu, X., Ghahramani, Z., Lafferty, L.: Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions. In: Proceedings of the 20th International Conference on Machine Learning, pp. 912–919 (2003)
2. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Scholkopf, B.: Learning with local and global consistency. In: Advances in Neural Information Processing Systems, vol. 16, pp. 321–328. MIT Press, Cambridge (2004)

3. Zhu, X., Ghahramani, Z.: Learning from Labeled and Unlabeled Data with Label Propagation. Technical Report CMU-CALD-02-107 (2002)
4. Blum, A., Mitchell, T.: Combining Labeled and Unlabeled Data with Co-Training. In: Proceedings of the Eleventh Annual Conference on Computational learning theory, pp. 92–100 (1998)
5. Chen, J., Ji, D., Tan, C., Niu, Z.: Relation Extraction Using Label Propagation Based Semi-Supervised Learning. In: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, pp. 129–136 (2006)
6. Niu, Z., Ji, D., Tan, C.: Word sense disambiguation using label propagation based semi-supervised learning. In: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics, pp. 395–402 (2005)
7. Girju, R., Nakov, P., Nastase, V., Szpakowicz, S., Turney, P., Yuret, D.: SemEval-2007 Task 04: Classification of Semantic Relations between Nominals. In: Proceedings of the Workshop SemEval-2007, the 45rd Annual Meeting of the Association for Computational Linguistics, pp. 13–18 (2007)
8. Li, H., Matsuo, Y., Ishizuka, M.: Graph Based Multi-View Learning for CDL Relation Classification. In: Proceedings of the 3rd IEEE International Conference on Semantic Computing, pp. 129–136 (2006)