

## Improved Method for Relation Adaptation to Extract the New Relations Efficiently from Relational Mapping

Miss Pragati Sapate, Dr. Sanjay T. Singh

---

**Abstract:** Traditionally, the web contains different semantic relations. The information extraction focuses on pre-specified request from small set of text. The main task of information retrieval is to extract or organize the information items as well as representation and access storage of items. There are various methods available for relation extraction presented by different authors. The main step we are focusing is supervised relation extraction. The method we propose for relation extraction for adapting new relation with supervised relation extraction system. It is based on three major concepts called domain adaptation, relation extraction and transfer learning. Our proposed method uses combination of under-sampling majority class and oversampling minority class. This paper shows that combination of these two methods improves the classifier performance. We evaluate proposed method for relation extraction using different dataset which contains entities for different relation. Using this method we are going to improve the precision, recall, F-score rate of relations which helps to improve the accuracy of relation those are novel or newly adapted. To overcome challenges in relation extraction that novel entities and relations constantly appear on the web as it contains both structured and unstructured text on the web. Our experimental result shows that the proposed method achieves F-score rate of 69.18. Moreover, it outperforms the numerous methods to adapt new relation efficiently.

**Keywords:** Domain adaptation, Relation extraction, semantic analysis, Entities, Web mining.

---

### I. Introduction

The web contains different semantic relation exist between entities. Information retrieval deals with the representation, access storage, organization of information items. It is not possible to extract the relation from the dataset for which it is not trained. It helps to represent semantic relation that exists between two entities. The key task in natural language processing is to extract the information and that information is divided into three steps i.e. conference resolution, entity recognition and relation extraction. The relation extraction is done on source and target relation. The type of relation is said to be source relation which existing relation type on which the relation extraction is trained. The target relation is nothing but the novel relation type to which we must adapt the relation. The corpus is input to the open information extraction which is set of text. The web contains different information related to many real world entities. For example the person, organization, location, etc. The entities are related by different semantic relation. The information extraction aimed to satisfy small homogeneous corpora which requests specified semantic relation from training data, like extracting the location and the time of seminar from set of announcements. We then create manual extraction rule for target relation. The open information extraction tool is used as paradigm for the system which takes the input as corpus and extracts the large number of target relation which gives the efficient extraction. For describing documents related to particular person can expressed with the help of semantic relation which will improve coverage in information retrieval. The supervised machine learning algorithm improves the relation extraction. It is dependent on availability of data for target relation which we want to extract. However it is not possible to create training data manually for semantic relation that exists among different entities. The information extraction is the task of automatically extracting structured from unstructured machine readable documents. The goal of information extraction is to allow computation to be done on the previously unstructured data. The Named Entity Recognizer(NER) is used as subtask of information extraction that seeks to locate and classify elements in text into predefined categories such as names of persons, organization, location, expression of times, quantities, percentage etc. The supervised approach requires textual analysis that involves POS tagging and dependency parsing. The alternative is to use structured representation and define an appropriate similarity metric for the classifier. There are many other methods are available for relation extraction presented by different authors. The hand built patterns requires hand labeling patterns for each relation. But these type of patterns are domain dependent. It is difficult to create training data manually for every new type of relation. The accuracy of relation is less in such type of patterns. The bootstrapping method can be used on seed instances i.e. if you don't have enough annotated text to train on but do have some seed instances of the relation. It is also called semi-supervised relation extraction or pattern based relation extraction. The training data required is sensitive to the original set of seeds. It requires that we have seeds for each relation. The bootstrapping method does not give high precision. We need to tune so many parameters. Unlike, supervised relation extraction can achieve high accuracy at least for some relation if we have lots of hand labeled training data. The unsupervised

relation extraction might not always extract the relation of our interest. For avoiding this we use relational duality. It defines the relation into the way user want which can be represented using more than one lexical or syntactic pattern. The relational duality is used between the definitions of semantic relations for clustering of entity pairs. The key goal of information extraction is to retrieve the entities that are related to user query while it will retrieve as few non relevant entities as possible. Given a crawled set of web text, we identify all the different semantic relations that exist between entities mentioned in the corpus. The unsupervised relation extraction does not retrieve the relations which are novel. Before this method for extracting any relation the open information extraction tool was being used. The disadvantage of open information extraction is that it takes the input as small set of web text and homogeneous set only. In single data driven pass only it will retrieve all the relations so there is possibility of missing some of relations or it will not extract some of the novel relations which are new and user might be interested in that relations.

## **II. Literature Survey**

In this section we are presented the review of various methods introduced by different authors for the relation extraction.

In [3], authors M. Pasca et al. presented the bootstrapping methods to relation extraction are attractive because they require markedly fewer training instances than supervised approaches do. Bootstrapping methods are initialized with a few instances of the target relation. During subsequent iterations of the bootstrapping process, new extraction patterns are discovered and are used to extract new instances. But the quality of the extracted relations depends heavily upon the initial seeds provided to the bootstrapping system as described in [4] by Zornista Kozareva et al.

In [5], authors M. Banko et al. presented the new approach called Open Information Extraction (Open IE) is a domain-independent information extraction paradigm. Open IE systems are initialized with a few manually provided domain independent extraction patterns. Moreover, open IE systems attempt to extract all relations that exist in a corpus; users cannot specify in advance which relation types (targets) they want to extract. Therefore, it is not guaranteed that we will be able to extract instances for the target relation type in which we are interested.

In [6], authors Jing Jiang proposed the multi-task transfer learning method to train a relation extraction system. She models commonality among different relation types by a shared weight vector. Next, a multi-class logistic regression model is trained using both source and target relations. To determine which features to share between relation types, they propose an alternating optimization procedure as well as several heuristics. This method does not compute a projection of features among relation types.

Domain adaptation methods can be classified broadly into fully supervised [7], [8] and semi-supervised adaptation [9], [10] [1]. In the fully supervised scenario, we have labeled data for the source domain and also invest in labeling a few instances in the target domain whereas; the semi-supervised version does not assume the availability of labeled data from the target domain to use unlabeled data from the target domain. Domain adaptation methods first identify a set of common features in source and target domains and then use those features as pivots to map source domain features to the target domain [1]. However, relation adaptation differs from domain adaptation because, in domain adaptation, it is assumed that the class labels remain the same in both source and target domains, only the distribution of data is different whereas, in relation adaptation, the source and target relation types are considered to be different.

Transfer learning is intended to transfer knowledge learned from one or more tasks to a new task [1]. In [11] R.K. Ando and T. Zhang introduced the alternating structure optimization (ASO) framework; a learning algorithm is first trained on a set of auxiliary problems. The linear prediction vectors for those problems are arranged as a matrix. Next, Singular Value Decomposition is performed on this matrix to compute a lower dimensional mapping between the features. The working hypothesis in ASO is that by jointly learning a set of related problems (auxiliary problems); we can learn some useful information related to the structure of the data, which is useful when learning a new task [1]. Relation adaptation can be seen as a special instance of transfer learning, where the source relations act as auxiliary problems. We must transfer the structural knowledge about source relations to a target relation type. However, relation adaptation necessitates that we overcome the additional challenge of learning the target relation using only a few seed instances [1].

In [12] I. Dhillon discussed, spectral clustering of bipartite graphs have been studied and in this case it turns out to be producing co-clusters [1]. If we represent the vertices of one partite of the bipartite graph in rows of a matrix and the vertices of the other partite in columns (the matrix  $M$  in Section 2.5 is such an example), then spectral clustering of the original bipartite graph gives co-clusters for the newly formed matrix. If we translate this result back to our scenario where we have relation specific patterns in one of the partites and relation independent patterns in the other partite of the bipartite graph, then we retrieve clusters which group relation specific and relation independent patterns that are semantically similar [1]. The co-clusters can be considered as providing an alignment between relation specific and relation independent patterns. This enables

us to perform relation adaptation because relation specific patterns (features) in both source relations and the target relation can be first mapped to relation independent patterns and then train a classifier in this common (lower dimensional) feature space [1].

### **III. Related Work**

The unsupervised relation extraction system takes input as web document. The web crawler selects textual windows that contain two entities A and B in the web document. But this method has disadvantaged that the cost crawling, storing and preprocessing is high. The unsupervised relation extraction uses Open Information Extraction for extracting any relation between entity pair. It clusters the entity pair considering their semantic relation but it might not always extracts the relation type of interest. The other approaches are also available for relation extraction. For example, the hand built pattern which requires hand labeling patterns for each relations. The problem with hand built pattern is they are hard to maintain and it gives the less accuracy. The other method available is bootstrapping method. If you don't have enough annotated text to train on but you do have some seed instances of the relation. The seed instances are sensitive to the original set of seeds. It does not give high precision. Considering all these methods and their limitations we make use of supervised relation extraction system. We can achieve high accuracy at least for some relations, if we have lots of hand labeled training data. The relation extraction is nothing but structuring of the information on the web. In supervised relation extraction, first textual analysis is done which contains POS tagging of the sentence.

The pattern extractor extracts the features from the sentence as lexical and the syntactic pattern. In single pass extraction, all the lexical and syntactic patterns of the available relations and entity pair are extracted. They need to extract for input to the classifier. The alternative is to use the structural representation and define an appropriate similarity metric for the classifier. The supervised relation extraction system is trained to extract specific type of relations which it might not able to extract for which it has not been trained. The information extraction is the task of extracting structured from unstructured machine readable documents. The goal of information extraction is to allow computation to be done on previously unstructured data. The Named Entity Recognizer can also be used as the subtask of information extraction that seeks to locate and classify elements in text into predefined categories such as names of persons, organization, location, expression of time, quantities, percentage etc. Along with relation extraction, we need adapt new relations as well. For relation adaptation features like lexical-syntactic pattern are obtained. The features that occur for source relation might not occur for target relation. The solution to this is to learn lower dimensional mapping between source and target relations. The second problem we need to overcome is that number of training instances for target relation is less as compared to source relation. The proposed solution for this is to perform one sided under sampling to select subset of source relation training instances to train multi-class classifier. After this preprocessing, the dataset becomes imbalance. The methods proposed for learning from unbalanced dataset can be classified as under sampling and oversampling. The use of Synthetic Minority Oversampling Technique (SMOTE) makes dataset balanced. The combination of under sampling of majority class and oversampling of minority class is made which achieves better classifier performance.

### **IV. Proposed Approach And Design**

#### **4.1 Problem definition**

The relation extraction is, given a crawled corpus of web text identify all the different semantic relation that exists between entities mentioned in the corpus. For extracting any relation requires machine reading which acquires structured knowledge from unstructured text. In information retrieval, basic function of semantic relation is contributing to increase of precision and recall.

Semantic relation is the relation between concept or meaning. The concept can be expressed by terms or expression. In information retrieval, basic function of semantic relation is contributing to the increase of precision and recall. For finding semantic relation, we are using semantic cues which are nothing but the clues about sentence structure to determine the meaning of word. Considering large set of web text, it is costly to create training data manually for every type of relation which we want to extract. Instead of annotating large set of web text, it would be easy and cost effective to if we adapt some existing relation extraction system to new relation type using small set of training instances. The existing relation extraction system has been trained as source relation and the novel relation type to which we must adapt is called target relation. Before adapting new relation we must overcome three fundamental challenges. First, a semantic relation exist between two entities can be expressed using more than one lexical and syntactic pattern. Second, the types of relation are strongly dependent on application domain.

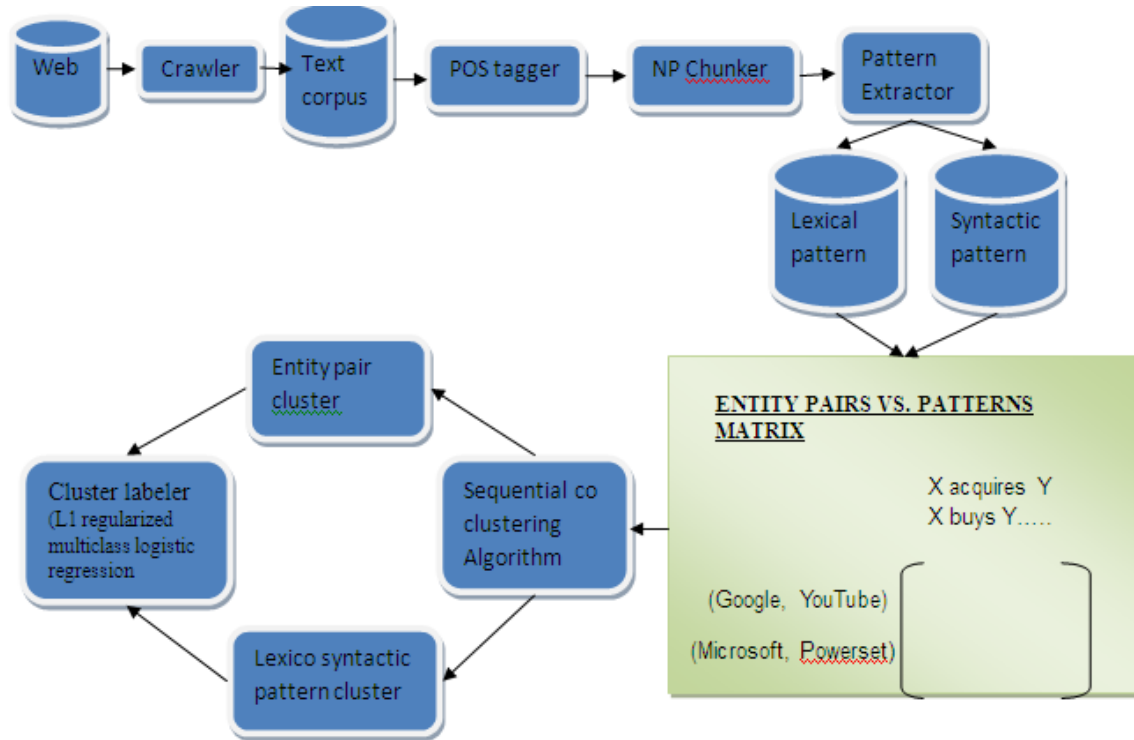


Fig 1: Process of relation extraction

In this paper, we have a tendency to use the quantity of various entity pairs from that two patterns are extracted because the edge weighting measure. We would like to point out that by constructing a simple bipartite graph and applying spectral clump techniques on that, we will accurately map patterns from source relations to the target relation.

**Relational Mapping:**

For relational mapping we use Relation Mapping algorithm. To find a lower dimensional mapping for patterns extracted from different relation types we use Spectral Graph theory. This lower dimensional mapping reduces the mismatches between patterns separated for source and the target relation. There are two inferences; first, if two vertices in a graph are connected to more common vertices, then those two vertices must be similar. Second, Two vertices are reciprocally similar if they are also similar within the original graph. In relation adaptation, we assume; first, if two relation-specific patterns are connected to many common relation-independent patterns, then those relation-specific patterns must be reciprocally similar. and third, there exist a lower dimensional latent space in which common patterns in the original space are located close together in this lower dimensional space.

**Relation mapping algorithm**

Input: M - edge-weight matrix of bipartite graph G, and Number of cluster k.

Process:

1. Evaluate an affinity matrix  $A \in \mathbb{R}^{n \times n}$  of G as  $A = \begin{bmatrix} 0 & M \\ M^T & 0 \end{bmatrix}$
2. Evaluate Laplacian as  $L = I - D^{-1} A$  where D is diagonal matrix. Elements of D are  $D_{ii} = \sum_j A_{ij}$  and I is unit matrix such as,  $I \in \mathbb{R}^{n \times n}$ .
3. Calculate an eigenvectors  $\vec{u}_1, \vec{u}_2, \dots, \vec{u}_k$ , for k smallest an eigenvalues of L and arrange them in columns to form the projection matrix  $U = [\vec{u}_1, \vec{u}_2, \dots, \vec{u}_k] \in \mathbb{R}^{n \times k}$
4. return compute the projection matrix U.

Output: A projection matrix,  $U \in \mathbb{R}^{n \times k}$ .

For relation classification we refer the One-sided under-sampling algorithm [1]. In this, to compensate for the loss of data because of imperfect feature projection, we compute new representation of entities. In relation adaptation, the number of target relation training entity pairs is appreciably smaller than that of the source relations. In imbalance dataset, most classifier considers the minority instances are used as noise and outlier.

However, relation adaptation differs from domain adaptation because, in domain adaptation, it is assumed that the class labels remain the same in both source and target domains, only the distribution of data is different whereas, in relation adaptation, the source and target relation types are considered to be different. Hence, understanding a classifier for a target relation kind that has solely many instances is tough to observe. We use a one-sided under-sampling algorithm to overcome this problem. We train classifier for balancing source relation and target relation. When the number of source relations are fewer than that of target relation. To overcome challenges in relation extraction that novel entities and relations constantly appear on the web as it contains both structured and unstructured text on the web. Our experimental result shows that the proposed method achieves F-score rate of 69.18. Moreover, it outperforms the numerous methods to adapt new relation efficiently. The proposed algorithm gives classified entities, but the dataset becomes imbalance which contains misclassified entities as well. To make dataset balance we implement SMOTE algorithm. The SMOTE improves accuracy of classifier of minority class.

#### **4.2 SMOTE algorithm**

The dataset becomes imbalance if the classification categories are not approximately equally represented. The under-sampling technique is used which attempts to select subset of training instances from the majority class whereas, oversampling technique synthetically generates instances from the minority class. The combination of under-sampling majority class and oversampling minority class improves the classifier performance. An oversampling approach is proposed in which minority class is oversampled by creating synthetic means instead of oversampling with replacement. In this method we need to create extra training data by performing certain operations on real data. We generate synthetic samples by using feature space rather than data space. Depending upon the amount of oversampling required neighbours are randomly chosen from k nearest neighbours. Our implementation uses five nearest neighbours. The synthetic samples are generated as, first take the difference between feature vector under consideration and its nearest neighbour. Then, multiply this number by random number between 0 and 1, and add it to feature vector under consideration. Algorithm is as follows.

Input: Number of minority class samples T; Amount of SMOTE N%; Number of nearest neighbors k

Output:  $(N/100) * T$  synthetic minority class samples

1. ( If N is less than 100%, randomize the minority class samples as only a random Percent of them will be SMOTEd. )
2. if  $N < 100$
3. then Randomize the T minority class samples
4.  $T = (N/100) * T$
5.  $N = 100$
6. endif
7.  $N = (\text{int})(N/100)$  ( The amount of SMOTE is assumed to be in integral multiples of 100.)
8. k = Number of nearest neighbors
9. numattrs = Number of attributes
10. Sample [ ] [ ]: array for original minority class samples
11. New index: keeps a count of number of synthetic samples generated, initialized to 0
12. Synthetic [ ] [ ]: array for synthetic samples  
(Compute k nearest neighbors for each minority class sample only. )
13. for  $i \leftarrow 1$  to T
14. Compute k nearest neighbors for i, and save the indices in the narray
15. Populate(N, i, narray)
16. endfor
- Populate(N, i, narray) ( Function to generate the synthetic samples. )
17. while  $N \neq 0$
18. Choose a random number between 1 and k, call it nn. This step chooses one of the k nearest neighbors of i.
19. for attr  $\leftarrow 1$  to numattrs
20. Compute:  $\text{dif} = \text{Sample}[\text{narray}[\text{nn}]][\text{attr}] - \text{Sample}[i][\text{attr}]$
21. Compute:  $\text{gap} = \text{random number between } 0 \text{ and } 1$



```

22. Synthetic[newindex][attr] = Sample[i][attr] + gap * dif
23. endfor
24. newindex++
25. N = N - 1
26. endwhile
27. return ( End of Populate. )
End of Pseudo-Code.

```

The use of SMOTE algorithm improves the performance of the classifier. The SMOTE provides new approach to oversampling. The combination of SMOTE and under-sampling performs better as compared to plain under-sampling. The results of SMOTE algorithm tested on the variety of datasets. The majority class is under-sampled by randomly removing the samples from majority class until the minority class becomes some specified percentage of majority class. For example, if suppose we under-sample the majority class at the rate of 200% then it would mean that the modified dataset will contain double the elements from the minority class as from the majority class.

## V. Implementation Details And Result

In this section IV, we are presenting practical environment, input dataset, metrics computed and performance graph.

### 5.1 Input Dataset

We use the data set that contains numerous relations that exist between entities of various varieties on the net. We use the Yahoo BOSS search API4 to transfer contexts for the entity pairs in the information set. Specifically, we tend to construct various contextual queries that embrace the two entities in associate degree entity pair and transfer snippets that contain those entities. The data set contains various relations that exist between entities of numerous types on the Web. An interesting research direction of relation adaptation is to extend the current method to handle entities that are not related as well as entities with multiple semantic relations. Moreover, in our proposed work we intend to apply the relation adaptation method in real-world relation extraction systems and evaluate its effectiveness in detecting novel relation types.

### 5.2 Result of practical work

The following graph shows the effect of number of relation independent patterns and how the precision, recall, F-score rate are changing. The supervised relation extraction method maintains precision and recall rate without losing accuracy of relations extracted. These values are being calculated depending upon the number of correctly classified entities.

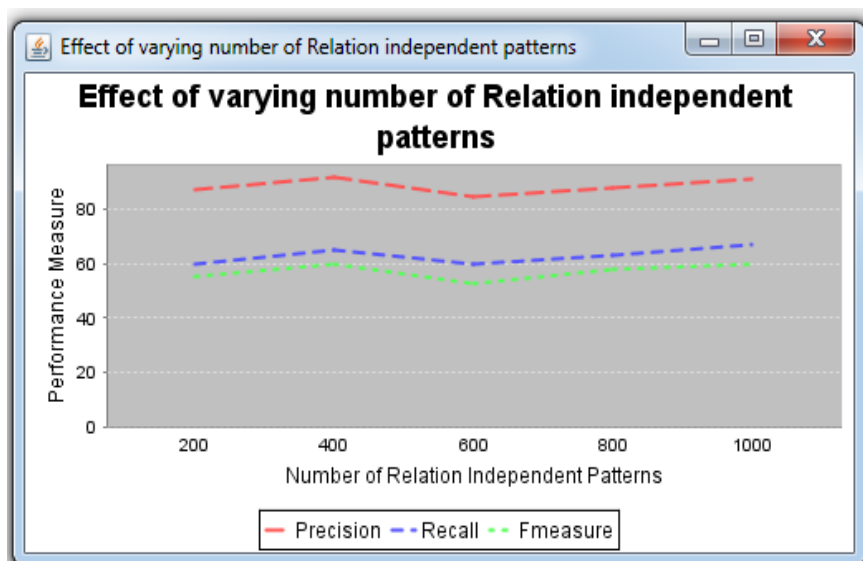
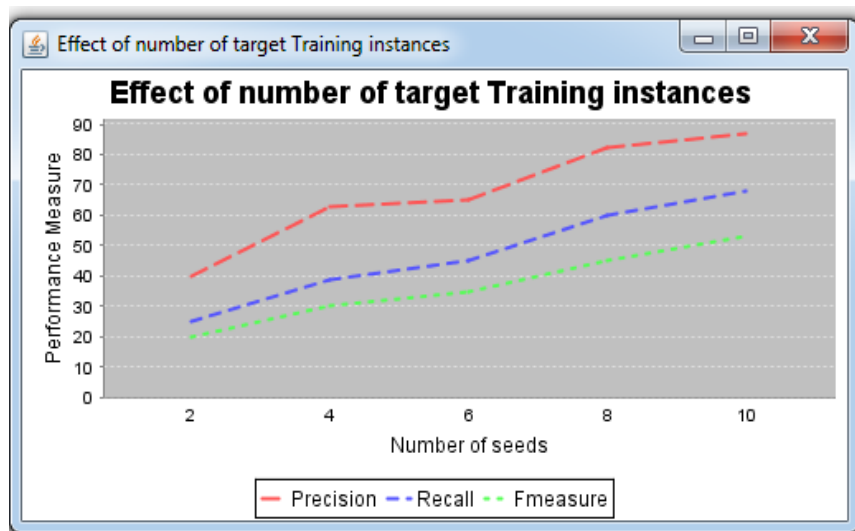


Fig 2: Effect of varying number of relation independent patterns



**Fig 3:** Effect number of target training instances

Both the graph shows that the performance increases as the number of target training instances increases. The result shows that the importance of target relations instances for relation adaptation. In our dataset, for each target relation we use multiple source relation.

## VI. Conclusion And Future Work

During this paper we have investigated and extended the efficient method for relation extraction with the aim of improving the precision rate, recall rate and the F-Score rate. The introduced is new architecture in which we have added the method for entities handling which was not present in previous methods. The entities those are either non-related or having multiple semantic meanings. Thus to avoid such dilemma and handle such entities we have added the entity handling. The results described here our practical work done so far over these algorithms. In the future work we will further evaluate this proposed framework completely under different real time environments and datasets. In this introduced is SMOTE algorithm is used in which oversamples are generated to remove non-stationary environment. And the results are comparing with existing one-sided-undersampling technique.

## Acknowledgement

We authors would like to thanks to Danushka Bollegala, Yutaka Matsuo, and Mitsuru Ishizuka for sharing their valuable knowledge.

## References

- [1]. Danushka Bollegala, Member, IEEE, Yutaka Matsuo, and Mitsuru Ishizuka, Member, IEEE, "Minimally Supervised Novel Relation Extraction Using a Latent Relational Mapping", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 25, NO. 2, FEBRUARY 2013.
- [2]. Danushka Bollegala Yutaka Matsuo Mitsuru Ishizuka, "Relation Adaptation: Learning to Extract Novel Relations with Minimum Supervision", Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence.
- [3]. M. Pasca, D. Lin, J. Bigham, A. Lifchits, and A. Jain. "Organizing and searching the World Wide Web of facts - step one: the one-million fact extraction challenge", In Proc. of AAAI'06, pages 1400–1405, 2006.
- [4]. Zornista Kozareva and Eduard Hovy, "Not all seeds are equal: Measuring the quality of text mining seeds" In NAACL'10, 2010.
- [5]. M. Banko, M. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni, "Open information extraction from the web", In IJCAI'07, pages 2670–2676, 2007.
- [6]. Jing Jiang, "Multi-task transfer learning for weakly-supervised relation extraction", In ACL'09, pages 1012–1020, 2009.
- [7]. H. Daume' III, "Frustratingly Easy Domain Adaptation," Proc. Ann. Meeting on Assoc. for Computational Linguistics (ACL '07), pp. 256-263, 2007.
- [8]. J. Jiang and C. Zhai, "Instance Weighting for Domain Adaptation in NLP," Proc. 45th Ann. Meeting on Assoc. for Computational Linguistics (ACL '07), pp. 264-271, 2007.
- [9]. J. Blitzer, M. Dredze, and F. Pereira, "Biographies, Bollywood, Boom-Boxes and Blenders: Domain Adaptation for Sentiment Classification," Proc. 45th Ann. Meeting on Assoc. for Computational Linguistics (ACL '07), pp. 440-447, 2007.
- [10]. S.J. Pan, X. Ni, J.-T. Sun, Q. Yang, and Z. Chen, "Cross-Domain Sentiment Classification via Spectral Feature Alignment," Proc. 19th Int'l Conf. World Wide Web (WWW '10), 2010.
- [11]. R.K. Ando and T. Zhang, "A Framework for Learning Predictive Structures from Multiple Tasks and Unlabeled Data," J. Machine Learning Research, vol. 6, pp. 1817-1853, 2005.
- [12]. I. Dhillon, "Co-Clustering Documents and Words Using Bipartite Spectral Graph Partitioning," Proc. Seventh ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD '01), pp. 269-274, 2001.

- [13]. F.R.K. Chung, Spectral Graph Theory, Regional Conf. Series in Math., Am. Math. Soc., 1997.
- [14]. I. Dhillon, "Co-Clustering Documents and Words Using Bipartite Spectral Graph Partitioning," Proc. Seventh ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD '01), pp. 269-274, 2001.
- [15]. M. Belkin and P. Niyogi, "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation," Neural Computation, vol. 15, no. 6, pp. 1373-1396, 2003.