Ranking-Based Automatic Seed Selection and Noise Reduction for Weakly Supervised Relation Extraction

Van-Thuy Phi[1], **Joan Santoso**[2], **Masashi Shimbo**[1] and **Yuji Matsumoto**[1,3]

[1] Nara Institute of Science and Technology
[2]Sekolah Tinggi Teknik Surabaya
[3]RIKEN Center for Advanced Intelligence Project (AIP)
[1]{*phi.thuy.ph8, shimbo, matsu*}*@is.naist.jp*
[2]*joan@stts.edu*

Abstract

This paper addresses the tasks of automatic seed selection for bootstrapping relation extraction, and noise reduction for distantly supervised relation extraction. We first point out that these tasks are related. Then, inspired by ranking relation instances and patterns computed by the HITS algorithm, and selecting cluster centroids using the K-means, LSA, or NMF method, we propose methods for selecting the initial seeds from an existing resource, or reducing the level of noise in the distantly labeled data. Experiments show that our proposed methods achieve a better performance than the baseline systems in both tasks.

1 Introduction

Bootstrapping for relation extraction (RE) (Brin, 1998; Riloff et al., 1999; Agichtein and Gravano, 2000) is a class of minimally supervised methods frequently used in machine learning: initialized by a small set of example instances called *seeds*, to represent a particular semantic relation, the bootstrapping system operates iteratively to acquire new instances of a target relation. Selecting "*good*" seeds is one of the most important steps to reduce *semantic drift*, which is a typical phenomenon of the bootstrapping process.

Another approach, called "*distant supervision*" (DS) (Mintz et al., 2009), does not require any labels on the text. The assumption of DS is that if two entities participate in a known Freebase relation, any sentence that contains those two entities might express that relation. However, this technique often introduces noise to the generated training data. As a result, DS is still limited by the quality of training data, and noise existing in positively labeled data may affect the performance of supervised learning.

In this study, we propose methods that can be applied for both automatic seed selection and noise reduction by formulating these tasks as ranking problems according to different ranking criteria. Our methods are inspired by ranking instances and patterns computed by the HITS algorithm, and selecting cluster centroids using K-means, latent semantic analysis, or the non-negative matrix factorization method. The main contributions of this paper are (a) an annotated dataset of 5,727 part-whole relations[1], which contains 8 subtypes for the bootstrapping RE system; (b) methods for automatic seed selection for bootstrapping RE and noise reduction for distant supervised RE; and (c) experimental results showing that the proposed models outperform baselines on two datasets.

2 Related Work

2.1 Automatic Seed Selection for Bootstrapping RE

As manually selecting the seeds requires tremendous effort, some research proposed methods to select the seed automatically. Eisner and Karakos (2005) used a "strapping" approach to evaluate many candidate seeds automatically for a word sense disambiguation task. Kozareva and Hovy (2010) proposed a method for measuring seed quality using a regression model and applied it to the extraction of unary semantic relations, such as"*people*" and "*city*". Kiso et al. (2011) suggested a HITS-based approach to ranking the seeds, based on Komachi et al. (2008)'s analysis of the Espresso algorithm (Pantel and Pennacchiotti,

[1]We release our annotated dataset at https://github.com/pvthuy/part-whole-relations.

Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 89–95
Melbourne, Australia, July 15 - 20, 2018.

2006). Movshovitz-Attias and Cohen (2012) generated a ranking based on pointwise mutual information (PMI) to pick up the seeds from existing resources in the biomedical domain. Given the seed set of a target relation, the goal of the bootstrapping method is to find *instances* similar to initial seeds by harvesting *instances* and *patterns* iteratively over large corpora, e.g., Wikipedia or ClueWeb.

2.2 Noise Reduction for Distantly Supervised RE

The DS assumption is too strong and leads to wrongly labeled data that affects performance. Many studies focused on methods of noise reduction in DS. Intxaurrondo et al. (2013) filtered out noisy mentions from the distantly supervised dataset using their frequencies, PMI, or the similarity between the centroids of all relation mentions and each individual mention. Xiang et al. (2016) introduced ranking-based methods according to different strategies to select effective training groups. Li et al. (2017) proposed three novel heuristics that use lexical and syntactic information to remove noise in the biomedical domain. The data generated by the noise reduction process can be used by supervised learning algorithms to train models.

3 Problem Formulation

Let R^* be the set of target relations. The goal is to find instances, or pairs of entities, upon which the relation holds. For each target relation $r \in R^*$, we assume there is a set D_r of triples representing the relation r. The triples in D_r have the form (e_1, p, e_2), where e_1 and e_2 denote entities, and p denotes the *pattern* that connects the two entities. A pair of entities (e_1, e_2) is called an *instance*. This terminology is similar to the one used in open information extraction systems, such as *Reverb* (Fader et al., 2011). For example, in triple $(\textit{Barack Obama}, \textit{was born in}, \textit{Honolulu})$, $(\textit{Barack Obama}, \textit{Honolulu})$ is the instance, and *"was born in"* is the pattern.

The two tasks we address are defined as follows:

Seed Selection for Bootstrapping RE: In automatic seed selection, a set R^* of target relations and sets of instance-pattern triples $D_r = \{(e_1, p, e_2)\}$ representing each target relation $r \in R^*$ are given as input. These triples are extracted from existing corpus or database, e.g., WordNet. With these inputs, the task is to choose *good* seeds from the instances appearing in D_r for each $r \in R^*$, such that they work effectively in bootstrapping RE.

Noise Reduction for Distantly Supervised RE: In noise reduction for distantly supervised RE, the input is the target relations R^* and the sets D_r of triples[2] generated automatically by DS for each relation $r \in R^*$. Because the data is generated automatically by DS, D_r may contain noise, i.e., triples (e_1, p, e_2) for which relation r does not actually hold between e_1 and e_2. The goal of noise reduction is to filter out these noisy triples, so that they do not deteriorate the quality of the triple classifier trained subsequently.

Formulation as Ranking Tasks: As we can see from the task definitions above, both seed selection and noise reduction are the task of selecting triples from a given collection. Indeed, the two tasks essentially have a similar goal in terms of the ranking-based perspective. We thus formulate them as the task of ranking instances (in seed selection) or triples (in noise reduction), given a set of (possibly noisy) triples. In the seed selection task, we use the k highest ranked instances as the seeds for bootstrapping RE. Likewise, in noise reduction for DS, we only use the k highest ranked triples from the DS-generated data to train a classifier. Note that the value of k in noise reduction may be much larger than in seed selection.

4 Approaches to Automatic Seed Selection and Noise Reduction

In this section, we propose several methods that can be applied for both automatic seed selection and noise reduction tasks, inspired by ranking relation instances and patterns computed by the HITS algorithm, and picking cluster centroids using the K-means, latent semantic analysis (LSA), or non-negative matrix factorization (NMF) method.

4.1 K-means-based Approach

The first method we describe is a K-means-based approach. It is described as follows: (**1**) De-

[2] To be precise, in each triple (e_1, s, e_2) generated by DS, s is not a pattern but a sentence that contains entities e_1 and e_2. However, we can easily convert each instance-sentence triple (e_1, s, e_2) to an instance-pattern triple (e_1, p, e_2) by looking for a pattern p that connects two entities in sentence s.

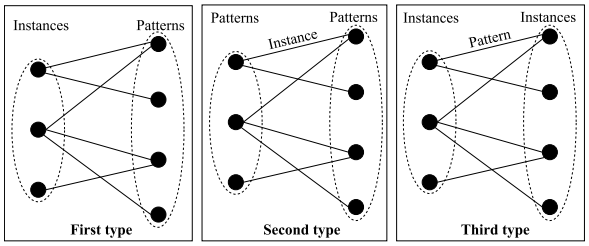

Figure 1: Graph representations of instances and patterns using the HITS algorithm.

termine the number k of instances/triples that should be selected[3]. (**2**) Run the K-means clustering algorithm to partition all instances in the input triples (see Section 3) into k clusters. Each data point is represented by the embedding vector difference between its entities; e.g., the instance $I = (\textit{Barack Obama}, \textit{Honolulu})$ corresponds to: $\text{vec}(I) = \text{vec}(\textit{"Barack Obama"}) - \text{vec}(\textit{"Honolulu"})$. We use pre-trained vectors published by Mikolov et al. (2013). (**3**) The instance closest to the centroid is selected in each cluster. Given that the number of clusters is k, the same number of instances/triples will be chosen.

4.2 HITS-based Approach

Hypertext-induced topic search (HITS) (Kleinberg, 1999), also known as the *hubs-and-authorities* algorithm, is a link analysis method for ranking web pages. In HITS, a good hub is a page that points to many good authorities and vice versa; a good authority is a page that is pointed to by many good hubs. These hubs and authorities form a *bipartite* graph, where we can compute the hubness score of each node.

In our task, let $\mathbf{A}$ be the instance-pattern co-occurrence matrix. We can compute the hubness score for each instance on the bipartite graph of instances and patterns induced by the matrix $\mathbf{A}$. Inspired by the way HITS ranks hubs and authorities, our HITS-based seed selection strategy can be explained as follows: (**1**) Determine the number k of triples that should be selected.(**2**) Build the bipartite graph of instances and patterns based on the instance-pattern co-occurrence matrix $\mathbf{A}$. Figure 1 presents three possible ways of building a bipartite graph. For the first type of graph, we consider each instance/pattern as a node in the graph. This representation is similar to that used by Kiso et al. (2011). In the second graph representation, patterns and instances are treated as nodes and edges, respectively. Similarly, instances and patterns are treated as nodes and edges, respectively in the last representation. (**3**) For the first and third types, we simply retain the top-k instances with the highest hubness scores as the outputs (we sort the instances in descending order based on their hubness scores). For the second type, k instances associated with the highest scoring patterns are chosen (we first sort the patterns in descending order based on their hubness scores).

[3] Depending on the task, instances or triples will be selected: instances for the automatic seed selection task, and triples for the noise reduction task. As instances are pairs of entities which are included in triples, we can simply convert between the instance and the triple, and apply a proposed method to both tasks.

4.3 HITS- and K-means-based Approach

In the combined method of HITS and K-means algorithms, we first rank the instances and patterns based on their bipartite graph and then run K-means to cluster instances in our annotated dataset. However, instead of choosing the instance nearest to the centroid, we retain the one that has the highest HITS hubness score in each cluster.

4.4 LSA-based Approach

Latent semantic analysis (LSA) (Deerwester et al., 1990) is also a widely used method for the automatic clustering of data along multiple dimensions. Singular value decomposition (SVD) is used to construct a low-rank approximation of the instance-pattern co-occurrence matrix $\mathbf{A}$. The SVD projection is performed by decomposing the matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ into the product of three matrices, namely an SVD instance matrix $\mathbf{I} \in \mathbb{R}^{M \times K}$, a diagonal matrix of singular values $\mathbf{S} \in \mathbb{R}^{K \times K}$, and an SVD pattern matrix $\mathbf{P} \in \mathbb{R}^{K \times N}$:

$$\mathbf{A} \approx \mathbf{ISP}^T$$

Our LSA-based seed selection strategy is as follows: (**1**) Specify the desired number k of triples. (**2**) Use the LSA algorithm to decompose the instance-pattern co-occurrence matrix $\mathbf{A}$ into three matrices $\mathbf{I}$, $\mathbf{S}$, and $\mathbf{P}$. We set the number of LSA dimensions to $K = k$. (**3**) We can consider LSA as a form of soft clustering, with each column of the SVD instance matrix $\mathbf{I}$ corresponding to a cluster. Then, we select the k instances that have the highest absolute values from each column of $\mathbf{I}$.

4.5 NMF-based Approach

Non-negative matrix factorization (NMF) (Paatero and Tapper, 1994; Lee and Seung, 1999) is an-

Subtype	Freq
Component-Of	643 (11.23%)
Member-Of	1,272 (22.21%)
Portion-Of	555 (9.69%)
Stuff-Of	1,082 (18.89%)
Located-In	534 (9.32%)
Contained-In	272 (4.75%)
Phase-Of	497 (8.68%)
Participates-In	872 (15.23%)
TOTAL	5,727 triples

Table 1: Statistics of our part-whole dataset.

Method	Average P@50
K-means	0.96
HITS_Graph1	0.90
HITS_Graph2	0.85
HITS_Graph3	0.90
HITS+K-means_Graph1	0.92
HITS+K-means_Graph2	0.85
HITS+K-means_Graph3	0.94
LSA	0.90
NMF	0.89
Random	0.75

Table 2: Performance of seed selection methods.

other method for approximate non-negative matrix factorization. The non-negative data matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ is represented by two non-negative factors $\mathbf{W} \in \mathbb{R}^{M \times K}$ and $\mathbf{H} \in \mathbb{R}^{K \times N}$, which, when multiplied, approximately reconstruct $\mathbf{A}$:

$$\mathbf{A} \approx \mathbf{WH}$$

The non-negativity constraint is the main difference between NMF and LSA. Similarly to the LSA-based method, we set the NMF parameter K to k, the desired number of instances to select. We then select the k instances that have the highest values from each column of $\mathbf{W}$.

5 Experiments

5.1 Datasets and Settings

We provide an annotated dataset of part-whole relations as a reliable resource for selecting seeds. Our dataset was collected from Wikipedia and ClueWeb, and annotated by two annotators. One of its special characteristics is that the part-whole relation is a collection of relations, not a single relation (Iris, 1989; Winston et al., 1987).

Table 1 gives the frequencies of each subtype of part-whole relations. There are 5,727 instances of 8 subtypes that were annotated with the same labels by both annotators. We use *Espresso+Word2vec* (Phi and Matsumoto, 2016), which is an improved version for the original Espresso algorithm (Pantel and Pennacchiotti, 2006). *Espresso+Word2vec* outperformed the Espresso system for harvesting part-whole relations by utilizing the *Similarity Ranker*, which uses the embedded vector difference between instance pairs of relations. The performance is measured with *Precision@N* (Manning et al., 2008), $N = 50$. In total, 5,000 instances are checked by annotators to ascertain whether they express part-whole relations. We vary the number k of seeds between 5 and 50 with a step of 5 to report the average P@50 of each seed selection method.

For the noise reduction task, we use the training and testing set developed by (Riedel et al., 2010), which contains 53 relation classes. This dataset was generated by aligning Freebase relations with the New York Times corpus. After removing noisy triples from the dataset using the proposed methods, we use the filtered data to train two kinds of convolutional neural networks (CNN) (the CNN model in (Zeng et al., 2014) and the PCNN model in (Zeng et al., 2015)) with at-least-one multi-instance learning (ONE) used in (Zeng et al., 2015), and the sentence-level attention (ATT) used in (Lin et al., 2016). Finally, we report the area under the precision-recall (AUCPR) of each noise reduction method.

5.2 Performance on Automatic Seed Selection Task

The performances of the seed selection methods are presented in Table 2. For the HITS-based and HITS+K-means-based methods, we display the P@50 with three types of graph representation as shown in Section 4.2. We use random seed selection as the baseline for comparison. As Table 2 shows, the random method achieved a precision of 0.75. The relation extraction system that uses the random method has the worst average P@50 among all seed selection strategies. The HITS-based method's P@50s when using Graph1 and Graph3 are confirmed to be better than when using Graph2. This indicates that relying on reliable instances is better than reasoning over patterns (recall that for the Graph2, we first choose

System	Original	+HITS	+LSA	+NMF	+Ensemble
CNN+ONE	0.180	**0.183**	0.173	0.178	0.181
CNN+ATT	0.234	0.235	0.235	0.233	**0.236**
PCNN+ONE	0.231	0.234	0.233	0.234	**0.235**
PCNN+ATT	0.248	0.253	0.250	0.252	**0.255**

Table 3: Performance (AUCPR) of each noise reduction method; in bold are the best scores.

the patterns, then select the instances associated with those patterns), as there is a possibility that a pattern can be ambiguous, and therefore, instances linked to that pattern can be incorrect. The K-means-based seed selection method provides the best average P@50 with a performance of 0.96. The HITS+K-means-based method performs better than using only the HITS strategy, while the LSA-based and NMF-based methods have a comparable performance.

5.3 Performance on Noise Reduction Task

Table 3 presents the performance of noise reduction methods. Recall that the K-means-based method achieves a high P@50 for the seed selection method. Our assumption is that each cluster may represent a set in which elements have similar semantic properties. However, we observed that as the number of relations is relatively high and there is no distinct definition between some relations in the distantly labeled data (e.g., the following three relations are quite similar: */location/country/capital*, */location/province/capital*, and */location/us_state/capital*, we decided not to perform the K-means-based method for our noise reduction task. The performances of the HITS-based, LSA-based, and NMF-based noise reduction methods are presented in Table 3. We experimentally set the portion of retained data from the distantly labeled data to 90%, given that the performance can be affected if too many sentences are removed from the original data. We also perform experiments with an ensemble method that combines the HITS-based and LSA-based strategies to merge rankings from their outputs, with half of the triples coming from the LSA-based method and the other half from the HITS-based method. Table 3 indicates that our proposed methods improved the performance of all CNN and PCNN models. Our ensemble method achieved the best improvements for three out of four systems, except that the HITS-based method obtained the best score for *CNN+ONE*.

6 Conclusion

We formulated the seed selection and noise reduction tasks as ranking problems. In addition, we proposed several methods, inspired by ranking instances and patterns computed by the HITS algorithm, and selecting clusters centroids using the K-means, LSA, or NMF method. Experiments demonstrated that our proposed methods improved the baselines in both tasks.

Acknowledgments

This work was partly supported by JST CREST Grant Number JPMJCR1513 and MEXT/JSPS Kakenhi Grant Number JP15H02749. We are grateful to the members of the Computational Linguistics Laboratory, NAIST and the anonymous reviewers for their insightful comments.

References

Eugene Agichtein and Luis Gravano. 2000. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the 5th ACM Conference on Digital Libraries*, pages 85–94. ACM.

Sergey Brin. 1998. Extracting patterns and relations from the world wide web. In *International Workshop on the World Wide Web and Databases*, pages 172–183. Springer.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391.

Jason Eisner and Damianos Karakos. 2005. Bootstrapping without the boot. In *Proceedings of the 2005 Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the 2011 Conference on*

Empirical Methods in Natural Language Processing, pages 1535–1545. Association for Computational Linguistics.

Ander Intxaurrondo, Mihai Surdeanu, Oier Lopez de Lacalle, and Eneko Agirre. 2013. Removing noisy mentions for distant supervision. *Procesamiento del lenguaje natural*, 51.

Madelyn Anne Iris. 1989. Problems of the part-whole relation. In *Relational Models of the Lexicon*, pages 261–288. Cambridge University Press.

Tetsuo Kiso, Masashi Shimbo, Mamoru Komachi, and Yuji Matsumoto. 2011. HITS-based seed selection and stop list construction for bootstrapping. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 30–36. Association for Computational Linguistics.

Jon M. Kleinberg. 1999. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632.

Mamoru Komachi, Taku Kudo, Masashi Shimbo, and Yuji Matsumoto. 2008. Graph-based analysis of semantic drift in Espresso-like bootstrapping algorithms. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 1011–1020. Association for Computational Linguistics.

Zornitsa Kozareva and Eduard Hovy. 2010. Not all seeds are equal: Measuring the quality of text mining seeds. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 618–626. Association for Computational Linguistics.

Daniel D Lee and H. Sebastian Seung. 1999. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791.

Gang Li, Cathy Wu, and K. Vijay-Shanker. 2017. Noise reduction methods for distantly supervised biomedical relation extraction. In *SIGBioMed Workshop on Biomedical Natural Language Processing (BioNLP '17)*, pages 184–193. Association for Computational Linguistics.

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2124–2133. Association for Computational Linguistics.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011. Association for Computational Linguistics.

Dana Movshovitz-Attias and William W. Cohen. 2012. Bootstrapping biomedical ontologies for scientific text using nell. In *BioNLP: Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*, pages 11–19. Association for Computational Linguistics.

Pentti Paatero and Unto Tapper. 1994. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126.

Patrick Pantel and Marco Pennacchiotti. 2006. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 113–120. Association for Computational Linguistics.

Van-Thuy Phi and Yuji Matsumoto. 2016. Integrating word embedding offsets into the Espresso system for part-whole relation extraction. In *Proceedings of the 30th Pacific Asia Conference on Language, Information and Computation: Oral Papers*, pages 173–181.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Proceedings of the 2010 Joint European Conference on Machine Learning and Principles of Knowledge Discovery in Databases (ECML PKDD)*, pages 148–163. Springer.

Ellen Riloff, Rosie Jones, et al. 1999. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the 16th National Conference on Artificial Intelligence and the 11th Innovative Applications of Artificial Intelligence Conference (AAAI/IAAI)*, pages 474–479.

Morton E. Winston, Roger Chaffin, and Douglas Herrmann. 1987. A taxonomy of part-whole relations. *Cognitive Science*, 11(4):417–444.

Yang Xiang, Qingcai Chen, Xiaolong Wang, and Yang Qin. 2016. Distant supervision for relation extraction with ranking-based methods. *Entropy*, 18(6):204.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1753–1762. Association for Computational Linguistics.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2335–2344. Dublin City University and Association for Computational Linguistics.