Distant Supervision for Relation Extraction

Zhen Wang (v-zw)

September 15, 2013

1 Background

Relational facts provided by knowledge bases (KB) can significantly improve the performance of many NLP applications. Currently, the relations maintained by a KB are mainly tuples of two entities such as "Founded(Jobs, Apple)" which we call *relation instance*. To build a KB which is capable to cover worldly facts, an extractor which can discover relation instances from unstructured texts is required. We construct such an extractor by training a multi-labels classifier where input is sentences contain the given entity pair (features are calculated from these sentences) and this pair is classified into one of the pre-defined relation types (or "OTHER").

Distant supervision (DS) provdies a paradigm to heuristically label a large corpus without human labor for training a relation extractor [2]. Given

- Σ, a large corpus such as Wikipedia articles, NYT, etc.
- E, a set of entities mentioned in Σ.
- R, a set of relation types.
- Δ, a set of ground relation instances of relations in R.
- T, a set of entity types, as well as type signature $r(E_1, E_2)$ for relations.

where the last three elements are provided by KB such as Freebase, for any sentence $s \in \Sigma$ that contains a pair of entity (e_1, e_2), if $\exists r \in R$ such that $r(e_1, e_2) \in \Delta$, regard s as expressing relation r. Specifically, for each sentence $s \in \Sigma$, they use Stanford's named entity tagger to detect all named entities in s. Within these detected entities, they consider every pair such as (e_1, e_2). If $\exists r \in R$ such that $r(e_1, e_2) \in \Delta$, they extract both lexical and syntactic features from s with respect to $e1, e2$ and combine features extracted from *all* mentions of (e_1, e_2) to form a positive example (x_i, y_i) where $y_i = r$. If $\forall r \in R, r(e_1, e_2) \notin \Delta$, a negative example is constructed.

2 Related Work

Original DS leads to many false positive examples which may dilute the discriminative capability of useful features. For instance, consider sentences "Michael Jackson was born in Gary." and "Michael Jackson moved from Gary.", suppose "place_of_birth(Michael Jackson, Gary)" is included in adopted KB, since both of the two sentences contain the entity pair (Michael Jackson, Gary), features calculated from the two sentences are combined and labeled with "place_of_birth". Obviously, the second mention does not express the labeled relation and thus contributes bad pattern to the labeled relation. To alleviate such problem, Riedel et al. [3] relax original assumption to: if an entity pair participate in a relation, *at least one sentence* that mentions the pair might express that relaton. In their model, each entity pair has a random variable Y taking value from $R \cup \{\text{NA}\}$ and each mention of the entity pair has a binary random variable $\mathbf{Z}_i$ which is activated if this mention indeed expresses the value of Y.

Inspired by such multi-instance learning model, *multi-instance multi-label* model is proposed to handle relation overlapping. For example, "Founded(Jobs, Apple)" and "CEO-of(Jobs, Apple)" are not exclusive. The entity pair (Jobs, Apple) possesses both the two relations at the same time while Riedel et al. constrained each entity pair to only one relation (Y for each entity pair) which is not reasonable. In Hoffmann et al. [1]'s model, each entity pair has a $|R|$-dimensional random variable $\mathbf{Y}$ ($\mathbf{Y}^r = 1$ means the entity pair has relation r) instead of a single $Y = r$. The label of this level is aggregated from the labels of mention level (each sentence mentions the entity pair has $\mathbf{Z}_j$ taking values from $R \cup \{\text{none}\}$) through a deterministic OR operator. The deterministic OR operator reflects the at least one assumption. Surdeanu et al. [4] proposed a similar model. Besides of the at least one feature, they also aggregate the labels for entity pairs from labels of their mentions with features that reflects dependencies between different relations of mentions. Suppose all the mentions for "(Jobs, Apple)" in our corpus express the relation "CEO-of", since the weak supervision is provided through labels of entity pair $\mathbf{Y}$ and adopted KB activates both $\mathbf{Y}^{\text{CEO-of}}$ and $\mathbf{Y}^{\text{Founded}}$, the former model may make the mention level classifier associate some mention with "Founded" while the latter one may correctly classify all mentions and keep $\mathbf{Z}$ and $\mathbf{Y}$ consistent by observing that $\mathbf{Z}_j =$ "Founded" and $\mathbf{Z}_j =$ CEO-of are generated jointly in many cases.

Min et al. add one layer to previous 3-layer multi-instance multi-label model. The added layer ($\mathbf{l}$) models the relations of a entity pair. Different from modeling and providing supervision at the same layer, they allow the supervision $\mathbf{y}_i^r =$ Unlabeled but $\mathbf{l}_i^r =$ Positive. Specifically, for an entity pair which does not appear in any relation in KB, its supervision $\mathbf{Y} = \mathbf{0}$ strongly suggests that patterns extracted from mentions of the entity pair indicates "none". In this model, the supervision is relaxed. Patterns extracted from mentions of such entity pair also have opportunity to indicate a certain relation. In a word, the 4-layer multi-instance multi-label model tend to make better use of unlabeled cases.

As you can see, all the above approaches ignored the false negative produced by both assumption of DS and incompleteness of KB. Xu et al. [6] propose a novel labeling strategy which leverages the entity types. They divided sentences with respect to a relation r into three classes:

- $POS(r) = \{s \in \Sigma | s \text{ contains some } (e_1, e_2) \text{ where } r(e_1, e_2) \in \Delta\}$
- $RAW(r) = \{s \in \Sigma | s \text{ contains some } (e_1, e_2) \text{ of required types of } r\}$
- $NEG(r) = \{s \in \Sigma | s \text{ contains some } (e_1, e_2) \text{ of required types of } r \text{ where } \exists e_j \text{ such that } r(e_1, e_j) \in \Delta \text{ and } r(e_1, e_2) \notin \Delta\}$

Then a passage retrieval component (given a relation r as query, retrieve sentences that are most likely to express r) is trained with positive examples $POS(r)$ and negative examples $NEG(r)$. They use this component to select "relevant" passages (i.e., sentences) from $RAW(r)$ and then select entity pairs from these top-ranked sentences. Selected entity pairs are added into KB to alleviate its incompleteness. However, the entity pairs in $NEG(r)$ may be caused by incompleteness of KB itself. Besides, $POS(r)$ are labeled according to original DS assumption as well as entity type requirement which is not always reliable as the "(Micheal Jackson, Gary)" example showed.

Takamatsu et al. [5] pointed that 91.7% entity pairs appear only once in Wikipedia articles. In such case, the at least one assumption is equivalent to original DS assumption for those multi-instance learning approaches. Takamatsu et al. proposed a generative model to predict whether a pattern (they define a pattern as entity types as well as the sequence of words on the path of the dependency parse tree between the two entities such as "[Person] born in [Location]") express a target relation. Intuitively, in our corpus, the entity types in a pattern will be instantiated by both entity pairs that are covered by KB and entity pairs that are unknown. If the number of the former kind of entity pairs is much larger than the latter kind of entity pairs, the pattern is very likely to be associated with corresponding relations. Thus we can revise some wrong label of unknown relation instances.

3 Critical Factors

For simplicity, existing approaches made use of Stanford's named entity tagger which has only four broad entity types {person, location, organization, miscellaneous, none}. In fact, most relations connect two entities where the entities belong to specific entity types such as "compose(musician, song)". Besides, a relation can be expressed by various sentences but the entity types is much more stable which means that entity type is discriminative to distinguish relatinos. Someone may argue that named entity recognition (NER) is not easier than relation extraction and thus we can not treat NER as a subproblem of relation extraction. Indeed, how to associate an entity to appropriate entity type is coupled with relation extraction. Without the context, given that "Apple" is an organization and "Jobs" is a person, we will guess the entity pair is connected by relation "CEO-of". Inversely, given "product-of(ipad, Apple)", we will guess that "Apple" is a company rather than one kind of fruits. Generally speaking, the relaion between an entity and its type is *isA*. Given that "isA(e_1, E_1)" and "isA(e_2, E_2)", the most popular type of edges (i.e., relations) between hyponyms of E_1 and hyponyms of E_2 is a good choice for connecting e_1, e_2.

Feature space is extremely sparse because the lexical and syntactic features change in different mentions of different relations. This is one of the reasons why Xu et al. use coarse features for training passage retrieval component.

Original DS as well as those 3-layer multi-instance learning models suffer from incompleteness of KB. The false negative examples make it difficult to classify one mention to appropriate relation because sharing common features between negative examples and positive examples reduce the importance of these features which actually belong to positive examples. The above 4-layer models seperate week supervision provided by KB from true labels of an entity pair so that unlabeled pairs are allowed to have not existed relations which may be more consistent with their mention level labels. However, there is still no a reasonable way to describe the dependencies between observed supervision and the layer that models labels of entity pairs. Besides, as a multi-label classifier, for a certain relation, we actually regard all the other relations as well as the "OTHER" relation as negative examples for it. However, relations are not exclusive.

References

[1] R. Hoffmann, C. Zhang, X. Ling, L. S. Zettlemoyer, and D. S. Weld. Knowledge-based weak supervision for information extraction of overlapping relations. In *ACL*, pages 541–550, 2011.

[2] M. Mintz, S. Bills, R. Snow, and D. Jurafsky. 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: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics, 2009.

[3] S. Riedel, L. Yao, and A. McCallum. Modeling relations and their mentions without labeled text. In *Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer, 2010.

[4] M. Surdeanu, J. Tibshirani, R. Nallapati, and C. D. Manning. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 455–465. Association for Computational Linguistics, 2012.

[5] S. Takamatsu, I. Sato, and H. Nakagawa. Reducing wrong labels in distant supervision for relation extraction. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 721–729. Association for Computational Linguistics, 2012.

[6] W. Xu, R. H. Le Zhao, and R. Grishman. Filling knowledge base gaps for distant supervision of relation extraction.