Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 969053, 10 pages
http://dx.doi.org/10.1155/2015/969053

Research Article

Bias Modeling for Distantly Supervised Relation Extraction

Yang Xiang, Yaoyun Zhang, Xiaolong Wang, Yang Qin, and Wenying Han

Key Laboratory of Network Oriented Intelligent Computation, Harbin Institute of Technology Shenzhen Graduate School, Shenzhen 518055, China

Correspondence should be addressed to Yang Xiang; xiangyang.hitsz@gmail.com

Received 24 March 2015; Accepted 11 August 2015

Academic Editor: Chih-Cheng Hung

Copyright © 2015 Yang Xiang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Distant supervision (DS) automatically annotates free text with relation mentions from existing knowledge bases (KBs), providing a way to alleviate the problem of insufficient training data for relation extraction in natural language processing (NLP). However, the heuristic annotation process does not guarantee the correctness of the generated labels, promoting a hot research issue on how to efficiently make use of the noisy training data. In this paper, we model two types of biases to reduce noise: (1) *bias-dist* to model the relative distance between points (instances) and classes (relation centers); (2) *bias-reward* to model the possibility of each heuristically generated label being incorrect. Based on the biases, we propose three noise tolerant models: *MIML-dist*, *MIML-dist-classify*, and *MIML-reward*, building on top of a state-of-the-art distantly supervised learning algorithm. Experimental evaluations compared with three landmark methods on the KBP dataset validate the effectiveness of the proposed methods.

1. Introduction

With the explosion of web resources, traditional supervised machine learning, which relies on a small set of manually annotated training samples, may not be able to catch up with the up-to-date information needs. Likewise for relation extraction, a hot research issue in NLP predicts semantic relations for a pair of name entities.

DS (distant supervision/weak supervision) annotates a large scale of free text with relation mentions from existing KBs, providing a way to alleviate the problem of insufficient training data for relation extraction. DS initially assigns relation labels to sentences according to relation mentions when a sentence contains a certain pair of entities but does not care about whether it actually conveys the corresponding semantic relation. Therefore, the heuristic annotation process does not guarantee the correctness of the generated labels, promoting a hot research issue on how to efficiently make use of the noisy training data.

For example, suppose a name entity pair ⟨`Obama, Hawaii`⟩ has three valid relation labels, `travel_to`, `born_in`, and `study_in`; according to the KB (i.e., Freebase), multiple sentences containing this entity pair from large-scale free text will be marked to convey either of the three relations (see S1–S3 in Table 1). However, we are not able to decide the specific relation label for each sentence in advance according to these annotations. Therefore, it is difficult for traditional supervised learning algorithms to learn directly from these heuristically annotated sentences. In addition, due to the incompleteness problem of either the KB or the free text, false negatives (FNs) and false positives (FPs) are inevitable, giving rise to noisy training labels. For example, at least one sentence in S1–S3 (Table 1) should be labeled with `study_in` according to the heuristics for DS, but actually none of them conveys this relation due to the incompleteness of free text. Similarly, because of the incompleteness of the KB, we can hardly find any relation from the KB that S3 can express.

Previous researches presented several methods to utilize the heuristically generated labels and train weak classifiers to predict unseen relations: single-instance learning (SIL) [1], multi-instance learning (MIL) [2], multi-instance multilabel learning (MIML) [3, 4] and some related extensions [5, 6], the embedding model [7] and the matrix factorization method [8, 9], and so on. Among them, multi-instance multilabel learning for relation extraction (*MIML-RE*) proposed by [4] is one of the state-of-the-art learning paradigms. MIML-RE treats multiple sentences (i.e., S1–S3) that contain a certain pair of

Table 1: Examples for DS annotated sentences.

	⟨Obama,Hawaii⟩	Latent label
Relations from the KB	travel_to ⟨O,H⟩ born_in ⟨O,H⟩ study_in ⟨O,H⟩	—
Annotated sentences	S1. President Obama will again travel to Hawaii for his annual holiday tradition.	travel_to
	S2. Born in Honolulu, Hawaii, Obama is a graduate of Columbia University and Harvard Law School.	born_in
	S3. Obama plays golf nearly every day while in Hawaii.	—

name entity as a bag (multi-instance) and has all possible labels (i.e., travel_to, born_in, and study_in) marked to these sentences (multilabel). Through modeling the distantly labeled data from the bag-level as well as the instance-level, MIML-RE transfers the data to styles that traditional supervised learning algorithms can easily deal with (see more details in Section 4).

Nevertheless, as mentioned above, the data generated by DS heuristics contain noises such as wrong labels; it is necessary to equip the learning algorithms with noise reduction methods. In addition, we argue that the *instance diversity problem* is prevalent for training weak classifiers. That is, we cannot guarantee that all the instances we collected and labeled through the DS heuristics are of high quality just like those labeled by human labelers.

The noisy problem can also be learned about from previous researches such as [10]. In [10], the authors manually annotated over 1800 sentences from free text and compared them to the KB. They finally got 5.5% FNs and 2.7% FPs, which is a good evidence for using noisy reduction strategies. Besides, from our observations toward the training data, we found that the instance diversity problem is remarkable in the dataset, reflected by the distributions for expectations (see Figure 2).

In this paper, we model two types of biases for noise reduction: (1) *bias-dist* to model the relative distance between points (instances) and classes (relation centers); (2) *bias-reward* to model the possibility of each heuristically generated label being incorrect. Bias-dist is modeled to weaken the *maximum probability assumption* (the class with the maximum probability should be assigned) during the EM process in MIML-RE, so that it is not always true that the class with the maximum probability is accepted. This bias is proposed according to the diverse qualities of training instances. Bias-reward is modeled to weaken the impact of wrong labels, resulting in the case that wrong labels would be with low predicting confidence. This bias aims at efficiently modeling the noisy group-level labels. Based on the biases, we propose three methods, *MIML-dist* (multi-instance multilabel learning with distance), *MIML-dist-classify* (multi-instance multilabel learning with distance for classification), and *MIML-reward* (multi-instance multilabel learning with reward), building on top of the MIML-RE framework. Therefore, this work can be seen as an extension of MIML-RE.

We set up experiments on one of the most popular benchmark datasets, the KBP dataset built by Surdeanu et al. [4]. Evaluation results compared with three landmark algorithms validate the effectiveness of the proposed methods. Particularly, MIML-dist-classify is built in the predicting phase, which is simple and fast to complete, boosting the $F1$ from the baseline 27.3% to 29.03%. MIML-dist-reward converges much faster than the original algorithm which reaches 29.01% on $F1$.

The contributions of this paper can be summarized as follows: (1) We are the first to explicitly model the bias related to the instance diversity problem and gain considerably better results; (2) the modeling methods toward the two types of biases are both validated to be efficient through the experiments.

The rest of the paper is organized as follows: Section 2 briefly introduces the literature; Section 3 describes the two types of biases; the models are detailedly described in Section 4. Sections 5 and 6 are the implementations and experiments. Discussion and conclusion are arranged at last.

2. Related Work

In this section, we briefly introduce the literature of distantly supervised relation extraction and the noise reduction methods for it.

2.1. Relation Extraction. Relation extraction (RE) is a hot research issue in NLP. In early researches, various approaches based on rich syntactic and semantic features were proposed. For example, Zelenko et al. introduced various subtree kernels with Support Vector Machine and Voted Perceptron learning algorithms [11]. In [12], the authors proposed three types of subsequence kernels for RE on protein-protein interactions and top-level relations from newspapers. Zhou et al. used tree kernel-based method with rich syntactic and semantic information and a context-sensitive convolution tree kernel [13]. Recent work focused mostly on deep neural network based structures, that is, single convolutional deep neural network based model [14, 15] and the combination of recursive neural network and convolutional neural network based model [16].

2.2. Distant Supervision for Relation Extraction. Distant supervision was firstly introduced in the biomedical domain by mapping databases to medical texts [17]. Since then, DS gained much attention in both information extraction (IE) and further RE. Most of the earlier researches include [18, 19] used single-instance learning according to the assumption that one pair of entity only corresponds to a single relation. In recent years, distant supervision is widely used in open IE to map Wikipedia infoboxes to wiki contents or web-scale texts [20, 21]. For RE, distant supervision is also employed for mapping Freebase relations to large scales of free text (i.e., New York Times) and predicting relations for unseen entity pairs [1–4]. Most aforementioned work used SIL, MIL, or MIML to train classifiers, which set strong baselines in this field. In addition, recent researches also include embedding based models that transferred the relation extraction problem into a translation model like $h + r \approx t$ [22–24], nonnegative matrix factorization (NMF) models [8, 9] with the characteristics of training and testing jointly, integrating active learning and weakly supervised learning [25], integer linear programming (ILP) [26], and so on.

2.3. Noise Reduction Methods for DS. One type of noise is that we cannot decide the actual label for each instance and can only estimate them according to some constrains. *At-least-one* is a representative constrain which considers that one relation label is positive when at least one of the mentions in the bag gets the label but discards the others. Related work directly modeled the noisy training data with multi-instance frameworks and learned model parameters through several times of EM iterations [2–4]. Intxaurrondo et al. [27] employed several heuristic strategies to remove useless mentions. Xu et al. [10] employed a passage retrieval model to expand the training data from instances with high confidence. Takamatsu et al. [28] directly model the patterns that express the same relation. Another type of noise is the wrong bag-level labels due to the incompleteness of either the KB or the textual corpus. Min et al. [5] put another layer to the MIML-RE architecture to model the true labels of a bag to model the incompleteness of the KB. Ritter et al. [6] added two parameters to directly model the missing of texts and the missing of KBs and set them with fixed values; they considered some side information such as popularity of entities as well. Fan et al. [9] added a bias factor b in their model to represent the noises. The idea of considering the instance diversity problem which relates to data quality in this work is a bit similar to Xu's passage retrieval model [10] but we are from a distinct perspective. The bias modeling idea is something like [6] but we model the missing in an indirect way which employs ranking-based measures.

3. Biases

In this section, we generally describe the two types of biases we propose.

3.1. Bias-Dist. Bias-dist (bias related to distance) aims at tackling the instance diversity problem rising from the DS annotation process. In traditional supervised machine learning, when human annotators label training instances, they incline to label those instances that they are confident of and discard the others so that a pure training set can be created. A typical example is the *annotation agreement standard* for evaluating a corpus and the instances whose labels are with hardly any disagreements are usually considered as being of high quality. On the contrary, there is no human intervention for the DS annotation; hence the quality of training samples cannot be guaranteed. As a result, when we use the classifier to assign relation labels to instances, it is likely that the predicting score for the true relation label is lower than that for a false label.

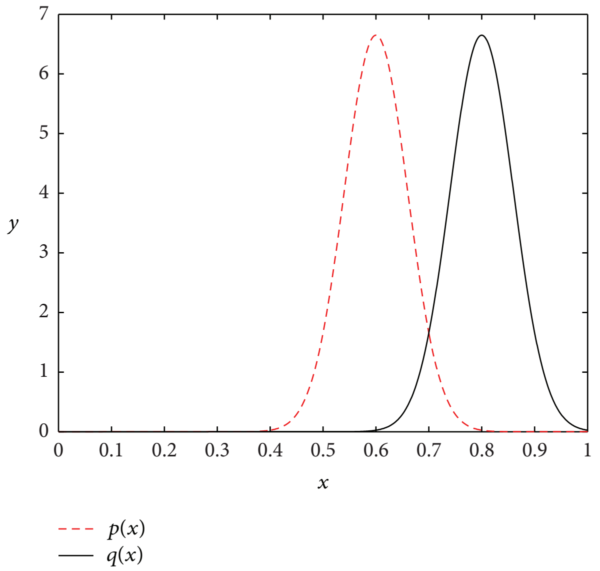


Figure 1: Gaussian distributions for illustrating the instance diversity problem.

More clearly, we assume that the predicting scores for the instances from two relation classes are drawn from two individual Gaussian distributions and show the case described above in Figure 1. Suppose $p(x)$ and $q(x)$ are two Gaussian distributions with expectations μ_p and μ_q ($0 < \mu_p < \mu_q < 1$), respectively. We further assume that the predicting scores for class p and class q follow these two distributions. The x-axis of Figure 1 denotes the predicting probabilities (scores) which range from 0 to 1. Suppose that a point t_i (i.e., $i = 0.8$) on the x-axis indicates a high probability (score) when predicting a certain instance m using a multiclass classifier. For both of the two classes, t_i may be an acceptable predicting score based on which we can classify the corresponding instance m into the positive areas for both of them. However, the distances from t_i to μ_p and μ_q reflect that distribution $q(x)$ has much stronger ability to generate t_i than $p(x)$. Thus in this case, according to the predicting score t_i and the mean for the two classes, the instance m should be classified to q rather than p.

To conclude, if the predicting scores for instances on different relation classes distribute diversely, the maximum probability assumption may not work well. Bias-dist is proposed to weaken this assumption through replacing the absolute predicting score to a relative form.

3.2. Bias-Reward. Bias-reward (bias according to label reward) is proposed to model the incompleteness of the KB and the textual corpus. The most typical setting for multi-instance learning in the literature is the *training bags*; that is, multiple instances containing a certain pair of name entities would fall into the same bag and share the same sets of labels.

As illustrated in Table 1, the bag-level labels come from the KB and would endure the incompleteness problem of the KB or free text. Further, many works [4, 29, 30] define bag-level positive labels as those from the KB and negative labels as the relations that the key entity (the first name entity in the pair) does not have according to the KB. As crucial constraints for distant supervision, noisy bag-level labels have bad effect on the models. Much formally, the constraints emphasize that

$$\begin{aligned}&\left(z_i^m = l : \forall l \in P_i, \exists m \in \mathbf{x}_i\right)\\&\wedge \left(z_i^{m'} = l' : \forall l' \in N_i, \neg\exists m' \in \mathbf{x}_i\right),\end{aligned} \tag{1}$$

where $\mathbf{x}_i$ stands for the ith bag, P_i and N_i denote the positive and negative label sets for this bag, and z_i^m is the relation label for the mth instance in the bag. If the KB is incomplete, the bag would have wrong negative labels. And if the textual corpus is not complete, the bag may be associated with wrong positive labels.

The problem of incompleteness is inevitable and has become a popular issue for distant supervision. We also take the entity `Barak Obama` as an example. If the KB we refer to is a year 2005's version but the free texts are recently collected, it is likely that the relation `president_of` ⟨`Obama, U.S.`⟩ is a wrong label and the sentences containing `Obama` and `U.S.` would be divided into the negative instances for the relation `president_of`.

To reduce the bad effects by incorrect negative labels, we add a reward to each bag-level negative label and multiply it by a weighting factor that reflects the likelihood of being nonnegative. Meanwhile, we add a penalty to each positive label and a weighting factor that reflects the likelihood of being nonpositive. We use a ranking-based method to determine the likelihood by computing rankings among all possible labels. More details would be described in Section 4.4.

4. Bias Modeling for Multi-Instance Multilabel Learning

In this section, we introduce the details of our methods for bias modeling.

4.1. Notations and Concepts. MIML takes a number of bags as the training data, learns a two-layer (the instance-level and the bag-level) weak classifier, and predicts relations for unseen sentences. For an easier description of MIML-RE and our methods, we define the following notations and concepts:

(i) $\mathscr{D}$, the whole textual corpus;

(ii) $\mathscr{R}$, the set of all known relation labels;

(iii) $\mathscr{L}$, the set of known relations for a certain entity (the first/key entity in a pair) from KB;

(iv) *instance, sample*, a sentence that contains the target entity pair and its quantized version for classification, respectively;

(v) *bag/group*, a set of instances that contain the same entity pair;

(vi) $\mathbf{w}_z$, the instance-level classifier (z-classifier), a multiclass classifier;

(vii) $\mathbf{w}_y$, the bag-level classifier (y-classifier), a set of binary classifiers;

(viii) μ_k, the expectation/mean of the probability distribution on predicting scores for relation k;

(ix) σ_k, the variance of the probability distribution on predicting scores for relation k;

(x) x, an instance in the dataset;

(xi) $\mathbf{x}_i$, the ith bag;

(xii) P_i, the positive label set of the ith bag;

(xiii) N_i, the negative label set of the ith bag.

In addition, we use l, z, r, and y to denote class labels and τ, c, t, α, β, γ, η, and θ for predefined constants. Thus, the training data for MIML-RE is constructed as the following: multiple *instances* containing the same pair of entities constitute a *bag* $\mathbf{x}_i$, with all possible relations for the pair as its positive label set P_i and $\mathscr{L} \setminus P_i$ as the negative label set N_i.

4.2. MIML-Dist. We construct two individual models based on bias-dist: (1) *MIML-dist* adds bias-dist to the training steps of MIML-RE and updates the label assignment process (E-step); (2) *MIML-dist-classify* simply adds bias-dist in the testing step for predicting new sentences. Following MIML-RE, MIML-dist uses the maximum likelihood estimation (MLE) to model the whole training data (2) and the hard expectation maximization (EM) algorithm to learn the model parameters ($\mathbf{w}_y$ and $\mathbf{w}_z$) iteratively ((3)–(6)). Consider

$$\begin{aligned}p(\mathscr{D}) &= \prod_{i=1}^{n} p\left(\mathbf{y}_i \mid \mathbf{x}_i, \mathbf{w}_y, \mathbf{w}_z\right)\\&= \prod_{i=1}^{n} p\left(\mathbf{y}_i, \mathbf{z}_i \mid \mathbf{x}_i, \mathbf{w}_y, \mathbf{w}_z\right),\\p\left(\mathbf{y}_i \mid \mathbf{x}_i, \mathbf{w}_y, \mathbf{w}_z\right)&\\&= \prod_{m=1}^{M_i} p\left(z_i^m \mid \mathbf{w}_z\right) \prod_{r\in P_i \cup N_i} p\left(y_i^r \mid \mathbf{z}_i, \mathbf{w}_y^r\right).\end{aligned} \tag{2}$$

E-Step. For each instance in a bag, its label is decided by both the instance-level classifier and the bag-level classifier. One has

$$\begin{aligned}z_i^m &= \arg\max_z p\left(z \mid x_i^m, \mathbf{w}_y, \mathbf{w}_z, \mu_z\right)\\&= \arg\max_z p\left(z \mid x_i^m, \mathbf{w}_z, \mu_z\right) \prod_{r\in P_i \cup N_i} p\left(y_i^r \mid \mathbf{z}_i', \mathbf{w}_y^r\right),\end{aligned} \tag{3}$$

where $\mathbf{z}_i'$ denotes the bag labels in which the label for the current instance m has been updated by the z-classifier and x_i^m stands for the mth instance in the ith bag. Consider

$$\begin{aligned}\log p\left(z \mid x_i^m, \mathbf{w}_z, \mu_z\right) &= \gamma \log p\left(l \mid x_i^m, \mathbf{w}_z\right) \\ &\quad + (1-\gamma) \log p\left[p\left(l \mid x_i^m, \mathbf{w}_z\right) \mid \mu_z\right]^t,\end{aligned} \tag{4}$$

where μ_z is the mean value of the predicting scores for class z and

$$p\left[p\left(z \mid x, \mathbf{w}_z\right) \mid \mu_z\right] = \left[p\left(z \mid x, \mathbf{w}_z\right) - \mu_z\right]^t. \tag{5}$$

We assume that the predicting scores for each relation follow a Gaussian distribution and μ_z is its expectation. The variance σ_k had little effect on the result according to our early experiments so we discard it.

Then the model parameters are updated through the M-step by (6).

M-Step. Consider the following:

$$\begin{aligned}\mathbf{w}_z^* &= \arg\max_{\mathbf{w}} \prod_{i=1}^{n} \prod_{m\in M_i} p\left(z_i^m \mid x_i^m, \mathbf{w}\right), \\ \mathbf{w}_y^{r^*} &= \arg\max_{\mathbf{w}} \prod_{1\le i\le n, r\in P_i\cup N_i} p\left(y_i^r \mid \mathbf{z}_i^*, \mathbf{w}\right).\end{aligned} \tag{6}$$

The following two equations are used to infer the instance-level and bag-level labels through corresponding classifiers.

Inference. Consider the following:

$$\begin{aligned}z_i^{m^*} &= \arg\max_{z} p\left(z \mid x_i^m, \mathbf{w}_z\right), \\ y_i^r &= \arg\max_{\{0,1\}} p\left(y \mid z_i^*, \mathbf{w}_y^r\right).\end{aligned} \tag{7}$$

In the testing phase, similar to MIML-RE, MIML-dist also employs a *Noisy-or* model instead of *at-least-one* to avoid data sparsity.

Noisy-or Model. Consider the following:

$$\text{Noisy-or } (r)_i = 1 - \prod_{m\in M_i} \left[1 - p\left(z \mid x_i^m, \mathbf{w}_z\right)\right]. \tag{8}$$

We show MIML-dist in Algorithm 1 (the procedures of MIML-dist-classify and MIML-reward are similar except for several tiny steps). The expectation μ_k for each class k is computed and stored after each label update in training (1.6-1.7 in Algorithm 1).

```
(1) Training phase:
    (1.1) foreach iteration $e$ in $T$
    (1.2)     foreach instance $x$ in each bag $i$
    (1.3)         foreach label $l$ in $\mathscr{R}$
    (1.4)             $l = \arg\max_l p(l \mid x, \mathbf{w}_y, \mathbf{w}_z, \mu_l) = \arg\max_l p(l \mid x, \mathbf{w}_z, \mu_l) \times \prod_{r\in P_i\cup N_i} p(y_i = r \mid \mathbf{l}_i', \mathbf{w}_y^r)$
    (1.5)             if $l$ != $l_{\text{org}}$ then
    (1.6)                 $\mu_{\text{org}} \longleftarrow \frac{N_{\text{org}} \times \mu_{\text{org}} - p(l_{\text{org}} \mid \mathbf{w}_z, x)}{N_{\text{org}} - 1}$
    (1.7)                 $\mu_l \longleftarrow \frac{N_l \times \mu_l + p(l \mid \mathbf{w}_z, x)}{N_l + 1}$
    (1.8)             end if
    (1.9)         end foreach
    (1.10)        $z^* = \arg\max_z p(z \mid x, \mathbf{w}_z)$
    (1.11)        $y^{r*} = \arg\max_{\{0,1\}} p(y \mid \mathbf{z}_i^*, \mathbf{w}_y^r)$
    (1.12)    end foreach
    (1.13)    $\mathbf{w}_z^* = \arg\max_{\mathbf{w}} \sum_{i=1}^{n} \sum_{m\in M_i} \log p(l_i^{m*} \mid x_i^m, \mathbf{w})$
    (1.14)    $\mathbf{w}_y^{l*} = \arg\max_{\mathbf{w}} \sum_{i,l} \log p(y_i^l \mid z_i^*, \mathbf{w})$
    (1.15) end foreach
(2) Testing phase:
    (2.1) Predict the bag-level labels using Noisy-or model.
```

Algorithm 1: MIML-dist.

4.3. MIML-Dist-Classify.

Analogous to MIML-dist, MIML-dist-classify also assumes that the predicting score for each relation follows a Gaussian-like distribution. The difference between them is that MIML-dist-classify computes the expectations after all the training iterations and thus is much easier and simpler than MIML-dist. It normalizes the predicting probability in the testing phase by

$$\begin{aligned}\log p\left(l \mid x, \mu_l, \mathbf{w}_z\right) &= \tau \log p\left(l \mid x, \mathbf{w}_z\right) \\ &\quad + (1-\tau) \log\left[p\left(l \mid x, \mathbf{w}_z\right) - \mu_l\right]^t.\end{aligned} \tag{9}$$

When comparing MIML-dist with MIML-dist-classify, from the perspective of time complexity, MIML-dist computes *bias-dist* of each relation label for each instance, which costs much when the scale of the training data is large or labels are updated frequently. Moreover, the time complexity makes the parameter tuning process more difficult. Comparatively, MIML-dist-classify changes very little the original training process, and if the model is trained (i.e., MIML-RE), it need not be changed any more. The parameter tuning only locates in the testing phase which is simple and fast.

To conclude, MIML-dist-classify is a kind of parameter tuning strategy on the classification hyperplanes. It is efficient for distant supervision because the training data in this task suffer from the instance diversity problem much heavier than most other supervised learning tasks, the training data of which are carefully polished by annotators. It is very likely that the probability distribution for each relation class diversify from each other very much. MIML-dist is a more direct way to model bias-dist, in that it changes the label assignment strategy by considering the impact of data diversity. Compared with MIML-RE, MIML-dist lowers down the chances of trapping into local minimums and thus is expected to perform better than MIML-RE.

4.4. MIML-Reward. Different from the previous two methods which modify the probabilities produced by the z-classifier, MIML-reward updates the probabilities generated by the y-classifier. Concretely, the multiplication item in (3) is updated by bias-reward. As mentioned above, we import a ranking-based method to determine the likelihood of each bag-level label being incorrect and add a reward or penalty from the original probability. Formally, we define the following notions:

(i) For a positive label, l-l is potentially wrong if some irrelevant (negative or unlabeled) labels have higher ranks than l.

(ii) For a negative label, l'-l' is potentially wrong if it has a higher rank than some positive labels.

Moreover, we define K_l as the instance that has the maximum predicting confidence for label l (the key instance for l) in the bag and $R(l)$ as the number of labels that has a higher rank than l:

$$\begin{gathered} K_l = \arg\max_{m \in M_i} p\left(z_i^m = l \mid \mathbf{w}_z, x_i^m\right), \\ R(l) = \sum_{l' \in \overline{R}} I\left[p\left(K_{l'} \mid \mathbf{w}_z, \mathbf{x}_i\right) > p\left(K_l \mid \mathbf{w}_z, \mathbf{x}_i\right)\right] \end{gathered} \tag{10}$$

in which $I[t]$ is the indicator function (if $t > 0$, $I[t] = 1$) and $\overline{R}$ can be any label set. Intuitively, for a positive label l in a bag, the bigger $R(l)$ is, the more possible this label is wrong when setting $\overline{R}$ to be the nonpositive label set, while, for a negative label l', the smaller $R(l')$ is, the more possible this label tends to be wrong, when setting $\overline{R}$ to be the positive label set. We employ two constants, α and β ($\alpha > 0$, $\beta > 0$), to denote the intensity of the above tendencies and take them as a reward or penalty to a single label. The posterior probabilities at the bag-level are computed instead by (l represents a positive label and l' represents a negative label)

$$\begin{aligned} \log p\left(y_i = l\right) &= \log \left\{ p\left(y_i = l \mid \mathbf{z}_i, \mathbf{w}_y^l\right) \right. \\ &\quad - \frac{\alpha}{Z} \sum_{l' \in R \setminus P_i} I\left[p\left(K_{l'} \mid \mathbf{w}_z, \mathbf{x}_i\right) > p\left(K_l \mid \mathbf{w}_z, \mathbf{x}_i\right)\right] \\ &\quad \left. + \eta \right\}, \\ \log p\left(y_i = l'\right) &= \log \left\{ p\left(y_i = l' \mid \mathbf{w}_y^{l'}\right) \right. \\ &\quad \left. + \frac{\beta}{Z} \sum_{l \in P_i} I\left[p\left(K_{l'} \mid \mathbf{w}_z, \mathbf{x}_i\right) > p\left(K_l \mid \mathbf{w}_z, \mathbf{x}_i\right)\right] + \theta \right\}, \end{aligned} \tag{11}$$

where Z is the normalized factor which is set to be the number of irrelevant labels (for each positive label) or the number of positive labels (for each negative label). η and θ are smoothing factors.

To conclude, MIML-reward is proposed to alleviate the problem of noisy labels. As we can read from (3), the label assignment is partly contributed by the bag-level labels (the second multiplication item), which is built on the assumption that all the bag-level labels are correctly annotated. However, noisy labels are inevitable according to our previous analysis. The penalty and reward mechanism for bias-reward is to weaken the assumption, allowing that some labels could be wrong and can be discovered and considered during training. Similar ideas can be seen in [6, 8] who also took into account the incorrectness of bag-level labels.

5. Implementation Details

For a fair comparison, most of the settings in implementation follow MIML-RE including the number of training iterations T for EM (up to 8 times) and the number of folds F for cross validation to avoid overfitting ($F = 3$). The constants γ and τ were optimized on the developing set and were finally set to be 0.7 and 0.5 for MIML-dist and MIML-dist-classify, and the constant t was set to be 2 for both the two methods. The penalty and reward parameters α and β were set to be 0.2 and 0.2, respectively. For the smoothing parameters η and θ in MIML-reward, we simply set them to 0.01. In addition, we use the same features as MIML-RE which takes multiple syntactic and semantic z-level features and dependency based y-level features. In addition, we added bias-dist only on those positive labels but discarded the negative label NIL. We also sampled 5% negative examples for training.

6. Experiments

6.1. Dataset Description. We test on the KBP dataset, one of the benchmark datasets in this literature constructed by Surdeanu et al. [4]. The resources are mainly from the TAC KBP 2010 and 2011 slot filling shared tasks [25, 26] which contain 183,062 and 3,334 entity pairs for training and testing. The free texts come from the collection provided by the shared task, which contains approximately 1.5 million documents from a variety of sources, including newswire, blogs, and telephone conversation transcripts. The KB is a snapshot of the English version of Wikipedia. After the DS annotation, we finally got 524,777 bags including 950,102 instances for training. For testing, 200 queries (a query means a key entity) from the TAC KBP 2010 and 2011 shared tasks containing 23 thousand instances are adopted, in which 40 queries constitute the developing set. The relation labels include slots of person (*per*) and organization (*org*), and the total number of labels is 41.

6.2. Experiments. We will show the evaluation metrics, experiment results, and some observations from the data in this section.

6.2.1. Evaluation Metrics

P/R Curve. Following previous work, we report the stability of the algorithms by figuring P/R curves. A P/R curve is generated through computing precision and recall by selecting different proportions of the testing data. Generally, the higher the position of a P/R curve is in the figure, the more stable the corresponding algorithm is.

Final Precision, Recall, and F1. The metrics precision, recall, and $F1$ are used to evaluate the performance of the models on the whole testing dataset. And we denote them by *Final P*, *Final R,* and *Final F1* to distinguish them from other PRFs with part of the testing data.

To specify, the testing set has the same data format as the training set which is constituted by groups. And the above metrics are computed according to the KBP slot filling tasks [31, 32] (on the entity level) rather than sentential classification.

6.2.2. Expectations for Each Relation. To show the inspirations for proposing bias-dist, we computed the expectations (means) for each relation after initialization (before training epochs denoted by *mean_b*) as well as at the end of training (denoted by *mean_e*). The values were computed by averaging all the predicting scores for those instances that are classified to that relation. This process was carried out on MIML-RE to show the instance diversity problem that the algorithm may suffer from. We report the distributions of expectations with an error bar (Figure 2). In the figure, each circle denotes the average predicting expectation among all the training epochs for the relation corresponding to the x-axis, and the upper error and the lower error stand for the maximum and the minimum expectations during training. Thus, the uneven curve shows the diversities between relations. We see that the maximum average expectation is about 0.94 (index = 2, `per:date_of_birth`) but the minimum one is only 0.3 (index = 25, `org:members`). Since the z-classifier considers only the absolute predicting confidence (both in training and testing), it is likely that the actual relation label for an instance just gets a small predicting score. Hence, a relative predicting score is necessary due to the diversity.

Another interesting thing we observe is the upper and lower errors. The distance between the upper and lower error for one relation indicates the change of class center and members during training. We see that several relations have their predicting expectations almost unchanged during the whole training process. We guess one reasonable explanation is that the instances of these relations are indeed pure enough for classification, so that the labels for these instances may hardly change during EM.

6.2.3. Baselines. We compare our models with three baselines: *Hoffmann*, *Mintz++*, and *MIML-RE*. Hoffmann is one of the representative MIML-based algorithms which uses *deterministic-or* decision instead of relation classifiers and it also enables relation overlaps [3]. Minz++ [4] is a modified version of the original Mintz model [1] in which each mention is treated independently and multiple predictions are enabled by applying Noisy-or. The performance of Mintz++ significantly outperforms the original Mintz model. As to MIML-RE, we choose the better model (also named as MIML-RE in [4]), which contains a modified version of y-level features from at-least-one.

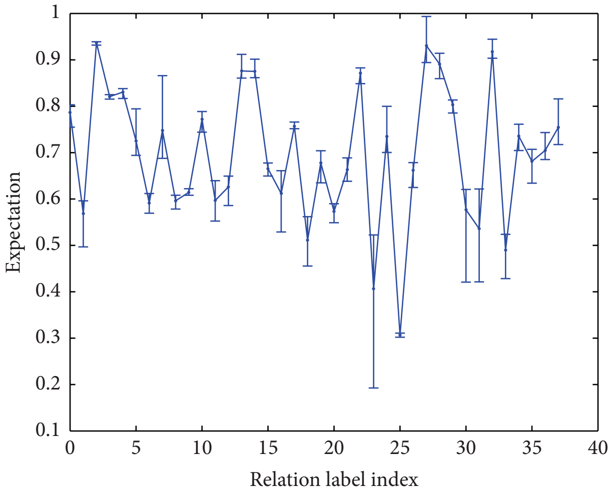


FIGURE 2: Diversities on expectations for different relations (relations that have no instances have been removed).

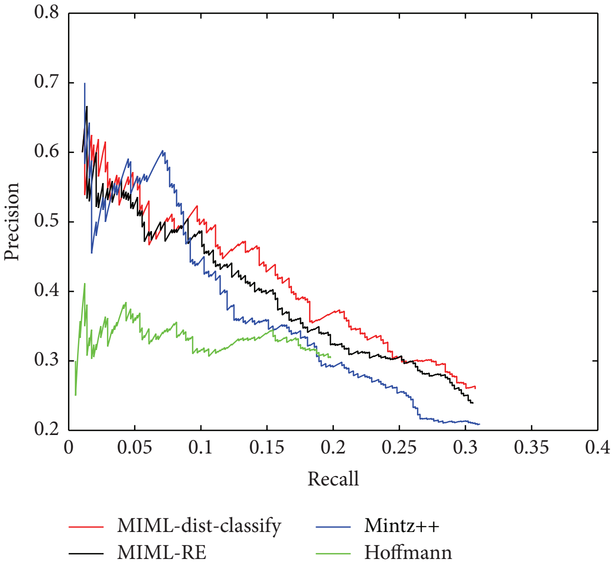


FIGURE 3: P/R curves of MIML-dist-classify and baselines.

6.2.4. Results. We firstly report the P/R curves of our proposed models compared with the three baseline methods mentioned above (Figures 3–5). The curves of the proposed methods are generated after tuning parameters on the developing set, aiming at maximizing Final $F1$.

For comparison, the best curve of MIML-dist-classify is tuned only based on the model generated by the last training epoch ($T = 8$). From Figure 3 we read that MIML-dist-classify has higher precision scores in both the low and the

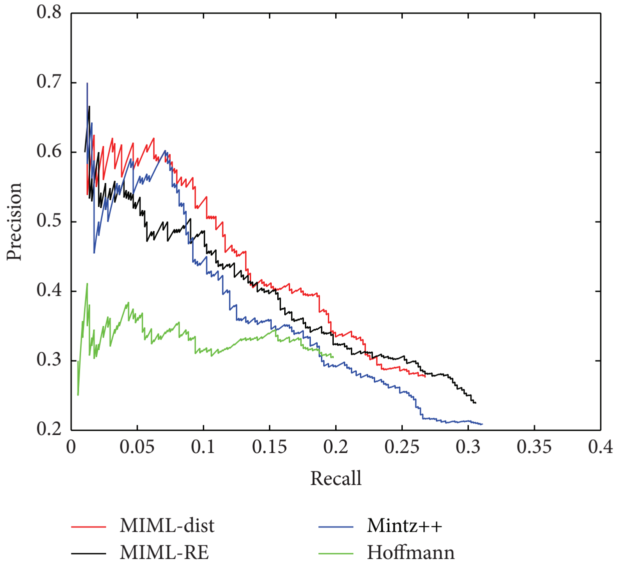


Figure 4: P/R curves of MIML-dist and baselines.

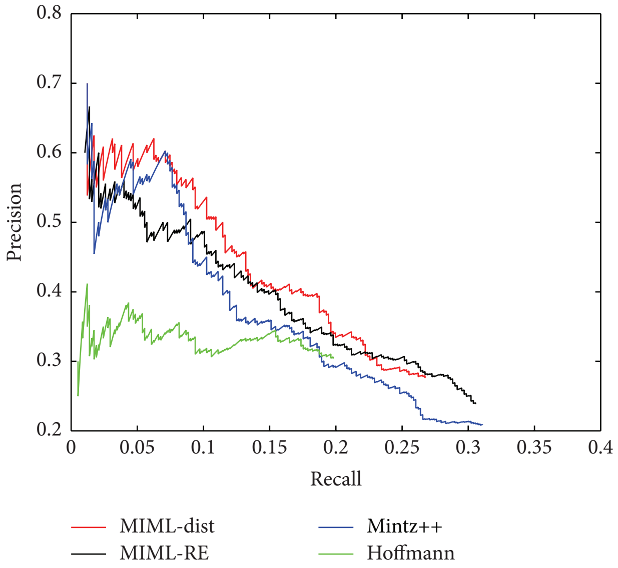


Figure 5: P/R curves of MIML-reward and baselines.

high recall proportions compared with the baselines but is worse than Mintz++ in the low recall proportion (0.05~0.1). However, the precision of Mintz++ drops down fast when recall goes beyond 0.1, not as stable as the other methods. Thus we conclude that although MIML-dist-classify is simple, it shows that bias-dist is beneficial to the final results.

MIML-dist has considerable good performance as MIML-dist-classify (Figure 4) especially in the low recall region (<0.15). We notice that when we fix the recall in 0.05–0.1, the precision of MIML-dist can be 5–10% points higher than MIML-RE. Other than MIML-dist-classify, the curve of MIML-dist has better overall performance than Mintz++.

Figure 5 shows the results generated by MIML-reward. We see that MIML-reward gains considerable improvements compared to MIML-RE in the low-recall region (<0.1) but falls beneath the model as the recall increases. Similar to MIML-dist, MIML-reward performs better than Hoffmann and Mintz++ almost over all the recall proportions.

Final P, Final R, and Final $F1$ are metrics that evaluate the methods on the whole testing set, which are also important performance measures in this literature. We can read from Table 2 that MIML-dist-classify improves the baselines by nearly 4% on recall while still keeping a relatively high precision. MIML-dist improves both precision and recall and achieves the maximum Final P among all the models. MIML-reward has the maximum Final R but its performance is at the cost of some precision points. We noticed that all the three methods we propose can enhance the baseline MIML-RE on $F1$ by over 1.5%. And compared with the other baselines, Hoffman and Mintz++, we observed that Final $F1$ is significantly improved by the proposed methods.

6.2.5. Case Study. We analyzed the predicted results of the proposed methods and compare them with those predicted by MIML-RE, which is a direct baseline of our work. To make it clear, we show in what kinds of cases our methods can make up the deficiency of the baseline.

Take one of the testing samples as an example: a sentence is predicted to `org:city_of_headquarters` with the probability of 0.56 and to the negative class label `NIL` with the probability of 0.43. According to the center of the positive class (0.82) which is far from 0.56, the sentence will not be predicted to the class any more after being normalized by bias-dist. We also figured that several positive predictions were directly replaced by the negative class after adding bias-dist, which is a contribution to the overall precision.

The effect of bias-reward can be indirectly read from the training bags to some extent since it depends on the bag-level labels which cannot be extracted from the testing set. According to the EM algorithm in MIML, the only supervision (weak supervision) is the bag-level labels, and the algorithm follows: if a label is positive in a bag, its ranking is higher than any other label. Hence, if the bag-level label is potentially wrong, it is likely that the algorithm falls into local minimums. We counted the number of different label assignments in each training epoch for MIML-reward and MIML-RE and found that it is really a large number (i.e., 352,192 different assignments in 950,102 when $T = 1$). We believe that this large number of differences can easily lead the training algorithm to converge to distinguishing directions.

Another thing we found is that the improvements distributed a bit evenly rather than focusing only on several specific relation labels. This indicates that the biases we propose are reasonable and efficient to all relations.

7. Discussion

We see that the proposed models work well on the whole testing dataset (Table 2) but from the P/R curves we realize (Figures 3–5) that the improvements on different proportions

Table 2: Best KBP2010 scores generated by the models.

	Final $P\%$	Final $R\%$	Final $F1\%$
Hoffmann	30.65	19.79	23.97
Mintz++	26.24	24.83	24.97
MIML-RE	30.56	24.68	27.30
MIML-dist-classify	29.60	28.47	**29.03**
MIML-dist	**31.42**	26.58	28.79
MIML-reward	27.20	**31.08**	29.01

of testing data are not so consistent, especially for MIML-dist and MIML-reward which perform much better at the low recall proportion but get a bit depressing when recall increases. We argue that there are several possible reasons: (1) the parameters (i.e., constants or biases) are tuned on the developing set to maximize the performance on Final $F1$ but not the P/R curve. So it is possible that other sets of parameters that do not perform well on Final $F1$ may generate a better curve (we indeed validated this through changing the parameter T); (2) the cases of each relation are a bit different that a fixed parameter toward all relation classes is not quite appropriate (i.e., α and β in MIML-reward); it is likely that the parameters only work well over all the testing set rather than some proportion. We need to further improve the learning algorithm so that more noises can be reduced or discarded. In addition, the hard EM training process suffers from the local minimum problem and how to tackle it should be further developed.

Another phenomenon we notice is that MIML-dist and MIML-reward have lower time complexity than MIML-RE. MIML-dist achieves the best result when $T = 6$ and MIML-reward gets the optimum when $T = 2$. It is believed that the biases especially bias-reward heavily change the label assignments so that the algorithm can converge much faster. As a result, we improve the time efficiency of the MIML algorithm. MIML-reward only needs 4-5 hours' running time, compared with MIML-RE whose training may last about 20 hours according to the authors.

Sometimes a simple method can achieve a good result, such as MIML-dist-classify, which only modifies the label assignment process in testing but boosts MIML-RE by 1.7% on $F1$. Besides, bias-dist can be applied in any probability classification model and bias-reward can also be integrated in any MIL framework which takes a bag as the basic training unit. However, we realize that there is still a long way for weak (distant) supervision to go since the results are still far behind what those supervised learning methods can achieve. Perhaps some more work can be down on either feature engineering or parameter selection.

8. Conclusion

In this paper, we propose three methods for distantly supervised relation extraction based on two types of biases. Among them, MIML-dist-classify and MIML-dist aim at tackling the instance diversity problem for different relations via adding bias items either in the testing step or in the training step. MIML-reward is introduced to model the bag-level label noise by adding rewards for wrong negative labels and penalties for wrong positive labels. Experimental results on a landmark dataset validate the effectiveness of the proposed methods, boosting Final $F1$ by 1.5%–1.7%. In the future, more flexible approaches would be researched to model the noises caused by DS.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported in part by National 863 Program of China (2015AA015405) and NSFCs (National Natural Science Foundation of China) (61402128, 61473101, 61173075, and 61272383). The authors thank the anonymous reviewers for their insightful comments.

References

[1] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, "Distant supervision for relation extraction without labeled data," in *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics*, August 2009.

[2] S. Riedel, L. Yao, and A. McCallum, "Modeling relations and their mentions without labeled text," in *Proceedings of the 2010 European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 148–163, 2010.

[3] R. Hoffmann, C. Zhang, X. Ling, L. Zettlemoyer, and D. S. Weld, "Knowledge-based weak supervision for information extraction of overlapping relations," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pp. 541–550, June 2011.

[4] M. Surdeanu, J. Tibshirani, R. Nallapati, and C. D. Manning, "Multi-instance multi-label learning for relation extraction," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pp. 455–465, July 2012.

[5] B. Min, R. Grishman, L. Wan et al., "Distant supervision for relation extraction with an incomplete knowledge base," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL '13)*, 2013.

[6] A. Ritter, L. Zettlemoyer, Mausam, and O. Etzioni, "Modeling missing data in distant supervision for information extraction," *Transactions of ACL*, vol. 1, pp. 367–378, 2013.

[7] J. Weston, A. Bordes, O. Yakhnenko, and N. Usunier, "Connecting language and knowledge bases with embedding models for relation extraction," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2013.

[8] S. Riedel, L. Yao, A. McCallum, and B. M. Marlin, "Relation extraction with matrix factorization and universal schemas," in *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 74–84, June 2013.

[9] M. Fan, D. Zhao, Q. Zhou, Z. Liu, T. F. Zheng, and E. Y. Chang, "Distant supervision for relation extraction with matrix completion," in *Proceedings of the 52nd Annual Meeting of the*

Association for Computational Linguistics, pp. 839–849, June 2014.

[10] W. Xu, R. Hoffmann, L. Zhao, and R. Grishman, “Filling knowledge base gaps for distant supervision of relation extraction,” in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL ’13)*, pp. 665–670, August 2013.

[11] D. Zelenko, C. Aone, and A. Richardella, “Kernel methods for relation extraction,” *The Journal of Machine Learning Research*, vol. 3, no. 6, pp. 1083–1106, 2003.

[12] R. C. Bunescu and R. J. Mooney, “Subsequence kernels for relation extraction,” in *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS ’05)*, pp. 171–178, December 2005.

[13] G. D. Zhou, J. Su, J. Zhang, and M. Zhang, “Exploring various knowledge in relation extraction,” in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL ’05)*, pp. 427–434, June 2005.

[14] D. Zeng, K. Liu, S. Lai, G. Zhou, and J. Zhao, “Relation classification via convolutional deep neural network,” in *Proceedings of the 25th International Conference on Computational Linguistics (COLING ’14)*, pp. 2335–2344, Dublin, Ireland, August 2014.

[15] C. N. dos Santos, B. Xiang, and B. Zhou, “Classifying relations by ranking with convolutional neural networks,” in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pp. 626–634, Beijing, China, July 2015.

[16] Y. Liu, F. Wei, S. Li, H. Ji, M. Zhou, and H. Wang, “A dependency-based neural network for relation classification,” in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, pp. 285–290, July 2015.

[17] M. Craven and J. Kumlien, “Constructing biological knowledge bases by extracting information from text sources,” in *Proceedings of the 7th International Conference on Intelligent Systems for Molecular Biology*, pp. 77–86, Heidelberg, Germany, August 1999.

[18] R. C. Bunescu and R. J. Mooney, “Learning to extract relations from the web using minimal supervision,” in *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pp. 576–583, June 2007.

[19] K. Bellare and A. McCallum, “Learning extractors from unlabeled text using relevant databases,” in *Proceedings of the 6th International Workshop on Information Extraction on the Web*, 2007.

[20] F. Wu and D. S. Weld, “Autonomously semantifying wikipedia,” in *Proceedings of the 16th ACM Conference on Information and Knowledge Management (CIKM ’07)*, pp. 41–50, November 2007.

[21] O. Etzioni, M. Banko, S. Soderland, and D. S. Weld, “Open information extraction from the web,” *Communications of the ACM*, vol. 51, no. 12, pp. 68–74, 2008.

[22] A. Bordes, N. Usunier, A. Garcia-Durán, J. Weston, and O. Yakhnenko, “Translating embeddings for modeling multi-relational data,” in *Proceedings of the 27th Annual Conference on Neural Information Processing Systems (NIPS ’13)*, December 2013.

[23] Z. Wang, J. Zhang, J. Feng, and Z. Chen, “Knowledge graph embedding by translating on hyperplanes,” in *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, pp. 1112–1119, July 2014.

[24] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, “Learning entity and relation embeddings for knowledge graph completion,” in *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, Canberra, Australia, 2015.

[25] G. Angeli, J. Tibshirani, J. Wu, and C. D. Manning, “Combining distant and partial supervision for relation extraction,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP ’14)*, pp. 1556–1567, Doha, Qatar, October 2014.

[26] A. Nagesh, G. Haffari, and G. Ramakrishnan, “Noisy or-based model for relation extraction using distant supervision,” in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP ’14)*, pp. 1937–1941, Doha, Qatar, October 2014.

[27] A. Intxaurrondo, M. Surdeanu, O. L. de Lacalle, and E. Agirre, “Removing noisy mentions for distant supervision,” *Procesamiento del Lenguaje Natural*, vol. 51, pp. 41–48, 2013.

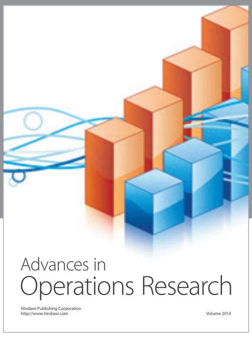
[28] S. Takamatsu, I. Sato, and H. Nakagawa, “Reducing wrong labels in distant supervision for relation extraction,” in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pp. 721–729, July 2012.

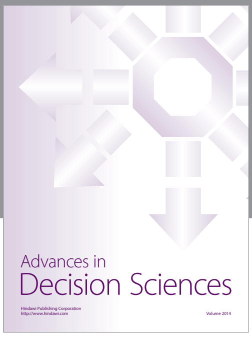
[29] A. Sun, R. Grishman, W. Xu, and B. Min, “New York University 2011 system for KBP slot-filling,” in *Proceedings of the 4th Text Analysis Conference (TAC ’11)*, Gaithersburg, Md, USA, November 2011.

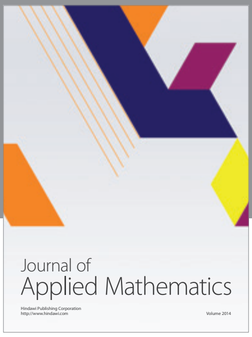
[30] M. Surdeanu, S. Gupta, J. Bauer et al., “Stanford’s distantly supervised slot-filling system,” in *Proceedings of the Text Analytics Conference*, 2011.

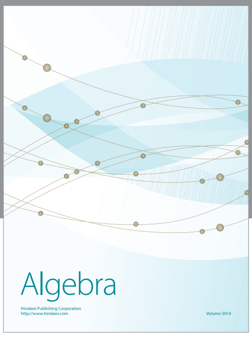
[31] H. Ji, R. Grishman, H. T. Dang, K. Griffitt, and J. Ellis, “Overview of the TAC 2010 knowledge base population track,” in *Proceedings of the Text Analytics Conference*, Gaithersburg, Md, USA, November 2010.

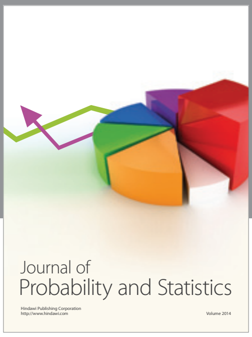
[32] H. Ji, R. Grishman, and H. T. Dang, “Overview of the TAC 2011 knowledge base population track,” in *Proceedings of the 4th Text Analysis Conference (TAC ’11)*, Gaithersburg, Md, USA, November 2011.











Hindawi

Submit your manuscripts at
http://www.hindawi.com