A Comparative Study of Relation Extraction Algorithms using Distant Supervision in Missing Data Models

A Thesis Submitted
in Partial Fulfilment of the Requirements
for the Degree of
Master of Technology

by

Saransh Srivastava
Roll No. : Y9317518

under the guidance of
Dr. Arnab Bhattacharya

Department of Computer Science and Engineering
Indian Institute of Technology Kanpur

May, 2015

CERTIFICATE

It is certified that the work contained in this thesis entitled **A Comparative Study of Relation Extraction Algorithms using Distant Supervision in Missing Data Models**, by **Saransh Srivastava(Roll No. Y9317518)**, has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

Arnab Bhattacharya

(Dr. Arnab Bhattacharya)
Department of Computer Science and Engineering,
Indian Institute of Technology Kanpur
Kanpur-208016

May, 2015

Abstract

Distant supervision algorithms learn relation extraction models using knowledge bases as the source of training. They are more suited for large corpora as they do not require any human annotation. The knowledge base most commonly used is Freebase which is a large semantic database of several thousand relations. Distant supervision algorithms combine the advantages of supervised and unsupervised information extraction, thereby improving the performance. However, most work till date has used heuristics such as a relationship not present in the database will not be present in text corpus. Such assumptions have led to a lot of missing data problem.

In this work, we will present two different approaches of handling missing data problem in distant supervision algorithm and after a comparative study, we will suggest which algorithm suits better for missing data problem. Both algorithms are improved versions of distant supervision algorithms for relation extraction without labelled data. The first algorithm proposes a latent-variable approach to model missing data. A local search approach is then used to inference, producing significant results. Moreover, it can be scaled to larger datasets. The second algorithm combines a passage retrieval model using coarse features into a relation extractor using multi-instance learning. In conclusion, we will discuss scenarios where both the algorithms produce better results than the present system.

Dedicated to

my parents and my nephew.

Acknowledgements

I would like to express my sincere gratitude to Dr. Arnab Bhattacharya, my thesis advisor, for his guidance and constant support during my thesis work. His regular discussion kept me motivated during the entire research and helped me formulate my thesis topic. His immense trust and faith in me kept me interested in the work. I am very grateful for his help and guidance during all this time. I thank the SigData students group for the regular group discussions and suggesting different ideas. I would also like to thank my dear friend Pavan Sharma for his patient listening and meaningful inputs in my research topic.

I would like to thank the Computer Science and Engineering department at IIT Kanpur for the excellent, hassle free environment for research, all the facilities and the outstanding academic training they provided during my stay here. The time spent with all my batch mates here has been memorable and I would like to thank everyone for their support.

I thank all my wing mates for all the fun they filled in my life spent at IIT Kanpur. Finally, I thank my parents and my two elder brothers for their constant support and belief in me.

Saransh Srivastava

Contents

List of Figures

Chapter 1

Introduction

Every day huge amounts of text data are uploaded on the web. The web pages are often filled with free form text, which is easy for humans to read but difficult for computers to understand. As we wish to share, remix and use this text information, such documents need to be structured and organized into tables, forms and other machine-readable formats. While most of the data is unique, there is a whole variety of data that can be clustered together, based on the relationships they commonly share.

Popescu [Pop07] characterises unstructured web text as redundant and having a broad coverage with multiple paraphrases. This text is written mostly in easy-to-understand language but with unreliable information and often ungrammatical language. In this thesis, we will present a novel approach of extracting relations from unstructured text corpus using *distant supervision learning algorithms*. In particular, we will discuss two implementations of distant supervision and then the problem of missing data in the knowledge base and text corpus, that these algorithms face. We will discuss how these implementations have partially overcome this problem. With experimental results we will be able to understand better, which implementation works better under certain conditions on the dataset.

1.1 Distant supervision learning for relation extraction

A *distant supervision learning algorithm* is a semi-supervised learning algorithm that applies a heuristic which assumes that each sentence which mentions the two related entities is an expression of a given relation. It uses a weakly labelled training set to supervise. The weakly labelled training set is a *knowledge base* which stores entity pairs based on relations they share. Knowledge base is a database which is expandable and inexpensive and, thus, a large amount of information can be stored. This is usually difficult with traditional supervised learning methods where training data is often hand labelled.

The idea is that any sentence which contains the pair of entities known in a knowledge base relation is likely to express that relationship in the sentence itself. There can be a lot of sentences containing the pair of entity, thus expressing the relationship. A classifier extract these sentences based on the relationship. Since these relations are structurally stored in the knowledge base, they are accurately mapped to the sentences. This avoids over fitting problem which may happen with unsupervised learning.

Distant supervised learning is known to perform better in relation extraction from unstructured text corpus compared to traditional methods like supervised and unsupervised learning algorithms. In unsupervised learning, large amount of data for learning can be used but the resulting relations extracted might not map to the given knowledge base relations. Distant supervision avoids these problems because of the presence of knowledge base.

1.2 Knowledge base as a source of labelled training set

A knowledge base is a centralized repository to store structured information used by a computer system. A well organized knowledge base consists of concepts, data, rules and specifications. In general, a knowledge base is not a static collection of information, but a dynamic resource that may itself have the capacity to learn.

Freebase [fre14] is currently the most popular and open source knowledge base used by academia. It is a large database of relations containing approximately 44 million topics and 2.4 billion facts (Wikipedia) and growing. Each individual pair of instances is linked by a relation called ‘relation instance’. For example, */people/person/place_of_birth* relation hold for entities named ‘Barack Obama’and ‘Honolulu’, so, Freebase has an instance (Barack Obama, Honolulu). Freebase contains data extracted from Wikipedia, NNDB (Notable Names Database - online database of biological details of over 40,000 people), FMD (Fashion Model Directory - online database of information about fashion models, fashion magazines, fashion designers and fashion editorials), MusicBrainz (structured open online database for music) and as well from individual contributors. For experimental purposes, binary instance representations were extracted, having more than 7,300 relations for 116 million instances between 9 million entities. Since Freebase has a lot of incomplete relation instances as well, all nameless entities were removed. Freebase has some reverse relations as well, which were merged (for example, person and place-of-birth versus place-of-birth and person).

However, with the introduction of a knowledge base as distant source of supervision, an inherent problem arises which is explained in the following section.

Person	**Employer**
Varun Sharma	Flipkart
Naveen Tiwari	InMobi
Pawan Kumar	IIT Delhi

True Positive	"**Varun Sharma**, a manager at **Flipkart** first came up with the idea of 'customer first' in business model."
False Positive	"**Naveen Tiwari** praised **inMobi** record revenue..."
False Negative	"**Pavan Sharma**, a professor at **IIT Kanpur's** Physics Department.."

Table 1.1: A hypothetical database and heuristically labelled training data

1.3 Missing data problem in relation extraction

Freebase though huge, is not complete. Thus, whenever there is a relationship which is not present in the database, it will treat it as a negative instance (false negative). For example, 93.8% of *persons* from Freebase have no *place of birth*, and as high as 98.8% of them have no *parent* information (Min et al. [MGW$^+$13]). Consider the example in Table 1.1 with a hypothetical data set.

Consider the pair of entity (Pavan Sharma, IIT Kanpur) is missing from the employer database then it is treated as a negative example of the relation. This is a major drawback as most databases of interest are highly incomplete. This gives us reason to extract information from the text corpus to extend the knowledge base.

1.4 Organization of the thesis

The rest of the thesis is organized as follows. Chapter 2 gives details about the previous work done in the area and other related works. We then describe in detail the two algorithms on distant supervision in Chapter 3. In Chapter 4, we will present an experimental comparative study of both the algorithms. Finally, in Chapter 5 we will conclude and discuss future work.

Chapter 2

Background work

Relation extraction from sentences has been an area of interest for a long time in the academia. Some of the early relation extraction algorithms are the DIPRE (Dual Iterative Pattern Relation Extraction) algorithm by Brin et al. [Bri99]. It uses string-based regular expressions that work on a semi-supervised learning technique. It exploits the duality between a set of patterns and relations to grow the target relation. Hearst et al. [Hea92] used a small number of regular expressions over words and part-of-speech tags to find examples of the hypernym relation. These patterns are replicated in systems, such as Etzioni et al. [ECD$^+$05].

Craven and Kumlien [CK$^+$99] were the first academicians who introduced distant supervision for information extraction. They used a yeast protein database as a knowledge base and extracted binary relations between proteins and cells/tissues/diseases/drugs. This approach recently became popular with mostly one or more approximations in learning. Wu et al. [WW07] enabled automatic info-box generation of Wikipedia articles by heuristically annotating Wikipedia articles which contain facts in the info-box. Benson et al. [BHB11] trained events extractor from Twitter by using music events taking place in New York city as a database for distant source of supervision.

Another work which specifically targets noise in heuristically labelled data generated by distant supervision is of Takamatsu et al. [TSN12] which present a generative model for the labelling process. This is used to pre-process, thereby improving the quality of labels before training the relation extractor. Min et al. [MGW$^+$13] extends the MIML model proposed by Surdeanu et al. [STNM12] using a semi-supervised approach assuming a fixed proportion of true positives for each entity pair. They further contributed in the analysis of the incompleteness of the Freebase knowledge base and the false negative match rate in two databases of labelled examples generated by distant supervision.

Another direction of work is the iterative semantic bootstrapping (Brin [Bri99]; Gravano and Agichtein [AG00]; Carlson et al. [CBK$^+$10]) which extracts lexical semantic resources from raw corpus by exploiting constraints between relations. At each iteration, relation entity tuples are evaluated and only the most reliable relations are kept for further iteration.

Chapter 3

Distant supervision algorithms

In this section, we will discuss the two algorithms which work on the principle of distant supervision for binary relation extraction. Then, we will present solutions to missing data problems in distant supervision algorithms which previous approaches (Riedel et al. [RYM10]; Hoffman et al. [HZL$^+$11]; Surdeanu et al. [STNM12]) ignored by heavily under-sampling the "negative" class. In the end, we will discuss the merits of the algorithms and their performance details.

3.1 Algorithm 1

In the given corpus, the task of the algorithm is to predict the relationship mentioned in each sentence or "NA" if the sentence does not mention a relationship. This will be then extended to find relationships which are missing in the database but are present in the corpus.

Assuming a set of sentences as: $s = s_1, s_2, s_3, ..., s_n$, which mentions a specific pair of entities (say, e_1 and e_2) and their corresponding latent sentence-level relationship mention variables, z = $z_1, z_2, z_3, ...z_n$. Latent sentence-level relationship mention variables indicate which relation is mentioned between e_1 and e_2 in each sentence. The task is to relate these latent sentence-level variables with aggregate binary variables d = $d_1, d_2, ...d_k$, which indicate whether

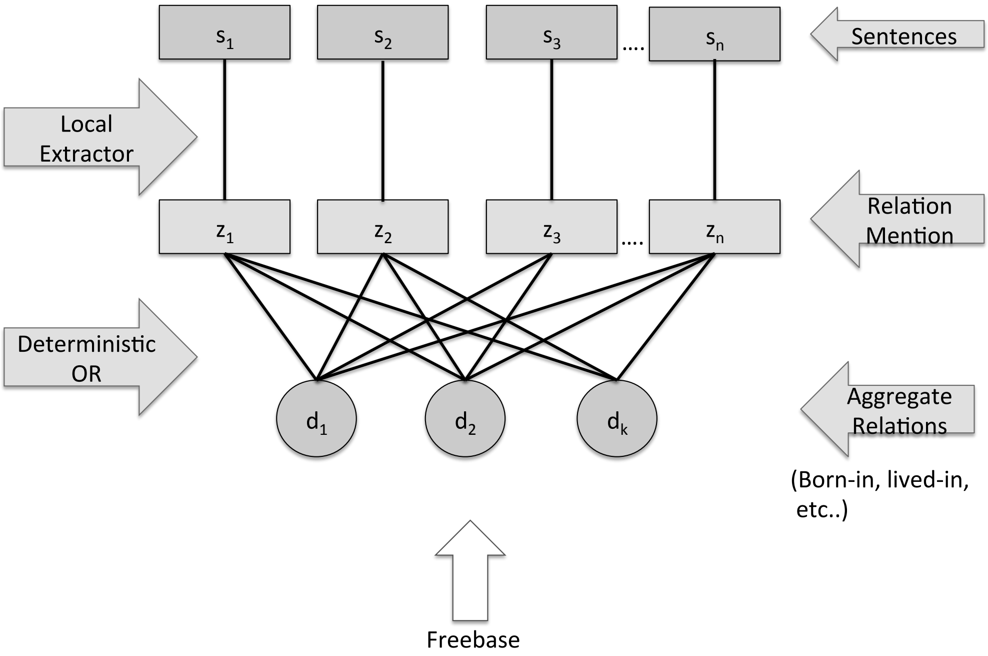


Figure 3.1: Relation extraction system

the preposition $r_j(e_1, e_2)$ is present in the database (Freebase). This is explained pictorially in Figure 3.1. Hoffmann et al. [HZL$^+$11] explains that a deterministic-OR function modelled over overlapping relations is a simple and effective choice for finding this relationship. A deterministic-OR states that if there exists at least one i such that $z_i = m$, then $d_m = 1$. The following example explains the point appropriately: Given two sentences in a corpus - "Bill Gates was the founder of Microsoft, Inc." and "Bill Gates was the CEO of Microsoft", then previous systems (Mintz et al. [MBSJ09]) assumes that relations do not overlap, thus both FOUNDER-OF (Bill Gates, Microsoft) and CEO-OF (Bill Gates, Microsoft) relation-entity pairs cannot exist together. Unfortunately, this in not true in most cases. However, if we consider any one of the relationship over the entire corpus, it is obviously true. Thus, instead of sentence level relations if we consider the relations $r_j(e_1,e_2)$ in aggregate, we can deduce whether the fact is true in the corpus by deterministic-OR. If none of the sentences mention the relation, then the fact is considered false.

3.1.1 Learning sentence-level relation mention

In this section, we will find a relation between aggregate-level variables, d_m, and sentence-level relation mentions, z_i. Now, in order to learn the sentence-level relation mention classifier θ, we will maximise the likelihood of the facts observed in Freebase conditioned on the sentences in our text corpus:

$$\theta^* \quad = \quad \arg \; \underset{\theta}{max} \; P(d|s;\theta) \tag{3.1.1}$$

By maximum likelihood estimation method for an independent and identically distributed sample:

$$P(d|s;\theta) = P(d_1, d_2, ..., d_k|s;\theta) \tag{3.1.2}$$

$$= P(d_1|s;\theta) \times P(d_2|s;\theta) \times ...P(d_k|s;\theta) \tag{3.1.3}$$

$$= \prod_{e_i,e_2} P(d|s;\theta) \tag{3.1.4}$$

In order to find the maximum likelihood, we will have to include all the relationship mention at sentence level for each entity pair:

$$\theta^* \quad = \quad \arg \; \underset{\theta}{max} \prod_{e_i,e_2} \sum_{z} P(d,z|s;\theta) \tag{3.1.5}$$

where d is the aggregate binary variable which indicates if the preposition $r_j(e_1, e_2)$ is present in the knowledge base, s is the set of sentences which contains the pair of entities (e_1, e_2) and z is the sentence-level relation mention variable. Now, in maximum-entropy taggers, the feature vector f together with sentence-level relation mention classifier θ, are used to define a conditional probability distribution over observed facts given sentences as:

$$P(d,z|s;\theta) \quad = \quad \frac{e^{\sum_i \theta.f(z_i,s_i)}}{C} \tag{3.1.6}$$

Here, C is a normalization constant ensuring proper probability distribution. Hoffmann et al. [HZL+11] included another term in the conditional probability which corresponds to the deterministic-OR function ω defined as follows:

$$\omega(z, d_j) = \begin{cases} 1 & if\ d_j = 1 \wedge \exists i : z_i = j \\ 0 & otherwise \end{cases} \tag{3.1.7}$$

Thus, the conditional likelihood of a given entity pair is defined as follows:

$$P(d|s;\theta) \quad = \quad \frac{1}{C}\prod_{i=1}^{n} e^{\theta.f(z_i,s_i)} \times \prod_{j=1}^{k}\omega(z, d_j) \tag{3.1.8}$$

where $\omega(z, d_j)$ factors are hard constraints corresponding to the deterministic-OR function and $f(z_i, s_i)$ is a vector of features extracted from sentences s_i and relation z_i. The expression is a Markov network expression. The model uses global normalization constant and this factor couples all of the parameters across network, preventing from decomposing the problem and estimating local groups of parameters separately. This global parameter coupling has significant computational ramifications, maximum likelihood parameter estimation with the complete data cannot be solved in closed form. Rather iterative methods, such as gradient ascent are used for optimization purpose.

In order to tune θ, an iterative gradient-ascent approach leading to a perceptron style additive (Collin et al. [Col02]) parameter update scheme, similar in style to the approaches of (Liang et al. [LBCKT06]; Zettlemoyer et al. [ZC07]) is used. The gradient of the conditional log likelihood, for a single pair of entities, e_1 and e_2 is as follows:

$$\frac{\partial log P(d|s;\theta)}{\partial\theta} = \mathbf{E}_{P(z|s,d;\theta)}\left(\sum_j f(s_j, z_j)\right) - \mathbf{E}_{P(d,z|s;\theta)}\left(\sum_j f(s_j, z_j)\right) \tag{3.1.9}$$

Now, we can define an update based on the gradient of the local log likelihood. However, these expectations are difficult to compute exactly and Hoffmann [HZL$^+$11] uses Viterbi approximation, by replacing the expectations with maximization for computation. Computing this approximation to the gradient requires solving two inference problems corresponding to the two maximization problems:

Most likely sentence extraction for the label facts, is

$$z^{*DB} \quad = \quad \arg \underset{z}{max} P(z|s, d : \theta) \tag{3.1.10}$$

and the most likely extraction for the input, without regard to the labels, is

$$z^{*} \quad = \quad \arg \underset{z}{max} P(z, d|s; \theta) \tag{3.1.11}$$

Now, predicting the most likely extraction without regard to labels can be done efficiently. Note that ω (in Eq: 3.1.7) represents deterministic dependencies between z and d, which when satisfied do not affect the probabilities of the solution. It is, thus, sufficient to independently compute an assignment for each sentence-level extraction variable z_i, ignoring the deterministic dependencies. The optimal solution for the aggregate variable d is then simply the assignment that is consistent with these extractions.

For Eq. (3.1.10), it is difficult to find the best assignment to the sentence-level hidden variables z = $z_1, ..., z_n$ conditioned on the observed sentences and facts in the database. Hoffmann et. al [HZL$^+$11] shows how this reduces to weighted edge cover problem. However, this learning is driven by hard constraints which causes a lot of false positives and false negatives while learning. The next section will present a modelling proposed by Ritter et al. [RZME13] to include such missing data.

3.1.2 Missing data modelling

The two assumptions corresponding to hard constraints are as follows:

1. If a fact is not found in the database it cannot be mentioned in the text.

2. If a fact is in the database, it must be mentioned in at least one sentence.

Thus, if there is data missing from either the text or database, it leads to errors in training data (false positives, and false negatives respectively).

Ritter et al. [RZME13] proposes to split the aggregate level variables, **d**, into two parts: **t** which represents whether the fact is mentioned in the text in at least one sentence, and **d'** which represents whether the fact is mentioned in the database. Pair-wise potential $\Psi(t_j, d_j)$ is introduced which penalizes disagreement between t_j and d_j:

$$\Psi(t_j, d'_j) = \begin{cases} -\alpha_{MIT} & if \quad t_j = 0 \quad and \quad d'_j = 1 \\ -\alpha_{MID} & if \quad t_j = 1 \quad and \quad d'_j = 0 \\ 0 & otherwise \end{cases} \tag{3.1.12}$$

Here, α_{MIT} (Missing In Text) and α_{MID} (Missing In Database) can be understood as penalties for missing information in text and database, respectively.

3.1.3 MAP inference

Since the new variable **t** which represents whether the fact is mentioned in the text in at least one sentence and **z**, which represents latent sentence-level relationship mention are deterministically related, finding a MAP solution to **z**, leads to the solution of **t**. With the introduction of **t**, the learning part explained in Section 3.1.1 now proceeds in similar fashion, with the exception that now the maximization is done over the additional aggregate-level variables **t**. Thus, each time when a fact is present in the text and not in database, -α_{MID} and when a fact present in database, +α_{MIT} is included.

The hard constraints are equivalent to setting α_{MID} = α_{MIT} = ∞, which is easier to infer than setting α_{MID} and α_{MIT} to fixed values as in Section 3.1.2.

So, in order to solve, greedy hill climbing method is proposed by Ritter et al. [RZME13]. It starts with full assignment of **z**, and repeatedly moves to the best neighbouring solution **z'**. To find the neighbour of **z**, we consider all relation-mention variables, z_i and in each iteration whichever z_i produces the largest improvement in the overall score, that z_i is considered as neighbour of **z**. When none of the neighbouring solutions have a higher score, then we accept that **z** as maximum and algorithm terminates. This maxima may or may not correspond to the global maxima, so random restarts are done to get the best local maximum.

3.2 Algorithm 2

This algorithm handles the problem of missing data in knowledge base with a different approach. Presented by Xu et al. [XHZG13], it combines a passage retrieval model using coarse features into a relation extractor using multi-instance learning.

3.2.1 Passage retrieval model

The passage retrieval component extracts coarse features of the document in order to provide complementary feedback to information extraction models. Unlike most other relation extraction models which exploit complex and rich features for information retrieval, this uses lexical features such as :

- Sequence of words between the entity pair
- Part-of-speech tags of these words
- A flag indicating which entity came first in the sequence

and syntactic features like:

- A dependency path between two entities

- For each entity, one ’window’ node that is not part of the dependency path

Xu et al. [RZME13] extracts two simple lexical features during the process: *Bag of Words* and *Word-Position*. For each relation r obtained, one binary classifier is independently trained. The combined model then predicts all relations for which the respective classifiers predicted positive results.

3.2.2 Pseudo-relevance relation feedback

It is an automatic local analysis method. A normal retrieval approach is applied to find an initial set of the most relevant documents and they are ranked. Top k ranked documents are assumed as most relevant and another relevance feedback is applied to the entire document under this assumption. The same approach is applied with the pair of entities and those relations which appear more in sentences are used to expand the database. The detailed algorithm is explained below.

Algorithm 1 Psuedo-relevance relation algorithm

Input: Set of ground facts of relations in R

Output: Relations corresponding to the top ranked entity pairs.

```
initialize $\Phi' \leftarrow \Phi$
for each relationship r $\epsilon$ R do
    learn a passage retrieval model P($r$)
      using coarse features and PDS(r) $\cup$ NDS(r)
      as training data
    score the sentences in the RDS(r) by P(r)
    score the pair of entities according to the scores
      of sentences they are involved in
    select the top ranked pair of entities, then add
      the relation $r$ to their label in $\Phi$’
end for
```

1. PDS(r) : Set of sentences in which the relation (mentioned in database) is expressed for any related pair of entities (positive data set).

2. RDS(r) : The rest of the dataset, which contains entities of the required

types in the knowledge base, e.g. for the relation *Founder-of* in Freebase, one person and one organization is mentioned.

3. NDS(r) : set of sentences where relationship does not hold for the pair of entities in the knowledge base but contains both primary and secondary entity.

4. R : set of relation names

5. Φ : set of ground facts of relations in R

After training the relation extraction model with the feedback model, the data set is passed through a state-of-the-art open-source system, MultiR (Hoffmann et al. [HZL$^+$11]) for relation extraction.

3.2.3 MultiR

MultiR, as first introduced by Hoffmann et al. [HZL$^+$11] is a probabilistic, graphical model of multi-instance learning handling overlapping relations. It is computationally tractable, producing accurate sentence-level predictions, decoding individual sentences as well as making corpus-level extractions.

Inference reduces it approximately to a weighted set cover problem. Iteration is done over the node and each time an edge having the highest weight incident and which does not violate the constraint is added. The worst case running time of the algorithm is $O(|R|.|S|)$ where R is the set of possible relations and S is the largest set of sentences for any entity pair.

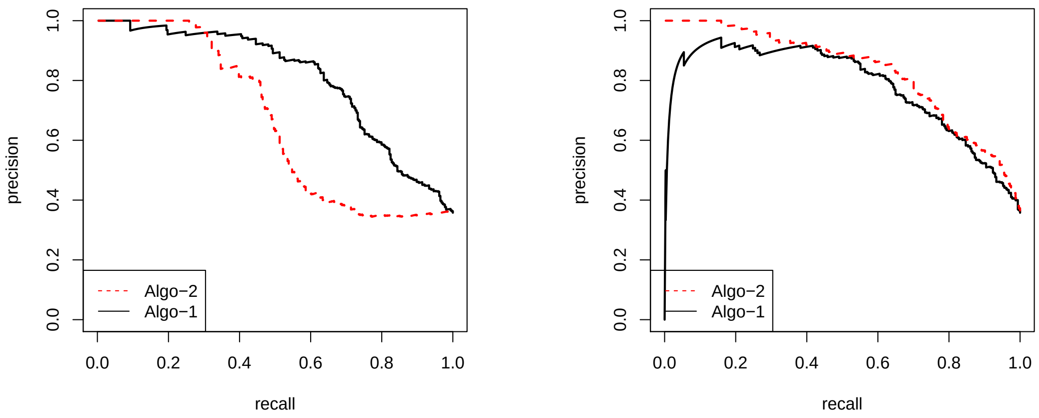
Chapter 4

Experiments

As presented in Chapter 3, both Algorithm 1 and Algorithm 2 are re-implementation of MultiR (Hoffmann et al. [HZL^{+}11]). For comparison purpose, we will do a detailed analysis of binary relation extraction for both the algorithms. The dataset used is New York Times text developed by Riedel et al. [RYM10], which contains approximately 1.8 million named entities collected over a span of 3 years and is aligned with the most popular knowledge base, Freebase.

Each algorithmic implementation extracts relation from the corpus and the common relations extracted are then used for comparison through precision recall values. The implementation is in Java and Scalala programming language, with the precision recall graphs implemented in R.

The entire set-up is run on an Intel core i7-4770 CPU and 3.40 GHz 64-bit processor with 32 GB RAM memory in Linux (Ubuntu 14.04) environment with maximum memory allocation pool size of 20G. In the next section, we will present the results obtained for each relation obtained. We will then present the sentential precision recall values over the entire data set and discuss why one algorithm performs better than the other.



(a) Without Passage-Retrieval Model (b) With Passage Retrieval Model

Figure 4.1: Overall Precision Recall Curve at sentence-level extraction.

4.1 Results

Since Algorithm 2 uses two types of lexical features in their passage retrieval system, we will compute our results on two types of data sets. First will be the normal data set without any preprocessing for lexical features for Algorithm 2. Figure 4.1a shows the overall precision recall curve at sentence-level extraction where Algorithm 1 has nearly 22% more area under the curve as compared to Algorithm 2. The graph shows that on low recall values both algorithms have high precision values but as recall increases Algorithm 1 performs better than Algorithm 2 indicating that on general data sets, without any preprocessing Algorithm 1 performs better than Algorithm 2.

Figure 4.1b shows the same curve with the data set preprocessed with lexical features extractor model. Here, Algorithm 2 performs better overall by 6.8% as compared to Algorithm 1. Precision recall curve increases its area under curve by 26% compared to the data set without passage retrieval model. At low recall values Algorithm 1 drops heavily on precision whereas Algorithm 2 maintains high precision as was also observed with the first dataset. With the increase in recall values, Algorithm 2 maintains higher precision compared to Algorithm 1, showing that when the dataset is pre-processed with coarse

Relation	AUC for Algorithm - 1	AUC For Algorithm - 2
business/company/founder	0.52	0.48
location/location/contains	0.87	0.75
people/person/children	0.86	0.85
business/person/company	0.95	0.91
location/neighbourhood/ neighbourhood-of	0.55	0.50
people/person/nationality	0.60	0.43
location/country/ administrative-division	0.021	0.029
location/us state/capital	0.0	0.0
people/person/place-lived	0.74	0.45
location/country/capital	0.006	0.005
people/deceased-person/ place-of-death	0.72	0.65
people/person/place-of-birth	0.39	0.35

Table 4.1: Per-relation AUC of PR curve

lexical feature extraction model, Algorithm 2 performs better than Algorithm 1 overall.

Table 4.1 shows the area under curve values for each relation extracted from the corpus corresponding to Figure 4.2. As shown by these values, when we run Algorithm 1 on the data set without any preprocessing, it performs better than the other for most cases. Relations like *location/location/contains*, *people/deceased-person/place-of-death* have more than 10% of coverage in Freebase, giving high AUC values and, thus, the gain from modelling missing data is not very significant. There are relations such as, *location/us-state/capital*, *location/country/administrative-division* and *location/country/capital*, which have less than 0.2% presence in Freebase and thus no useful overlap can be obtained.

There are some relations obtained by the algorithms which have more than 0.7% coverage in Freebase but shows high AUC values, such as *business/person/company*, *location/neighbourhood/neighbourhood-of* and *people/person/-place-lived*. Interestingly, for relation *people/person/place-lived*, Algorithm 1 has a precision recall curve value as 0.74 whereas Algorithm 2 has 0.45, show-

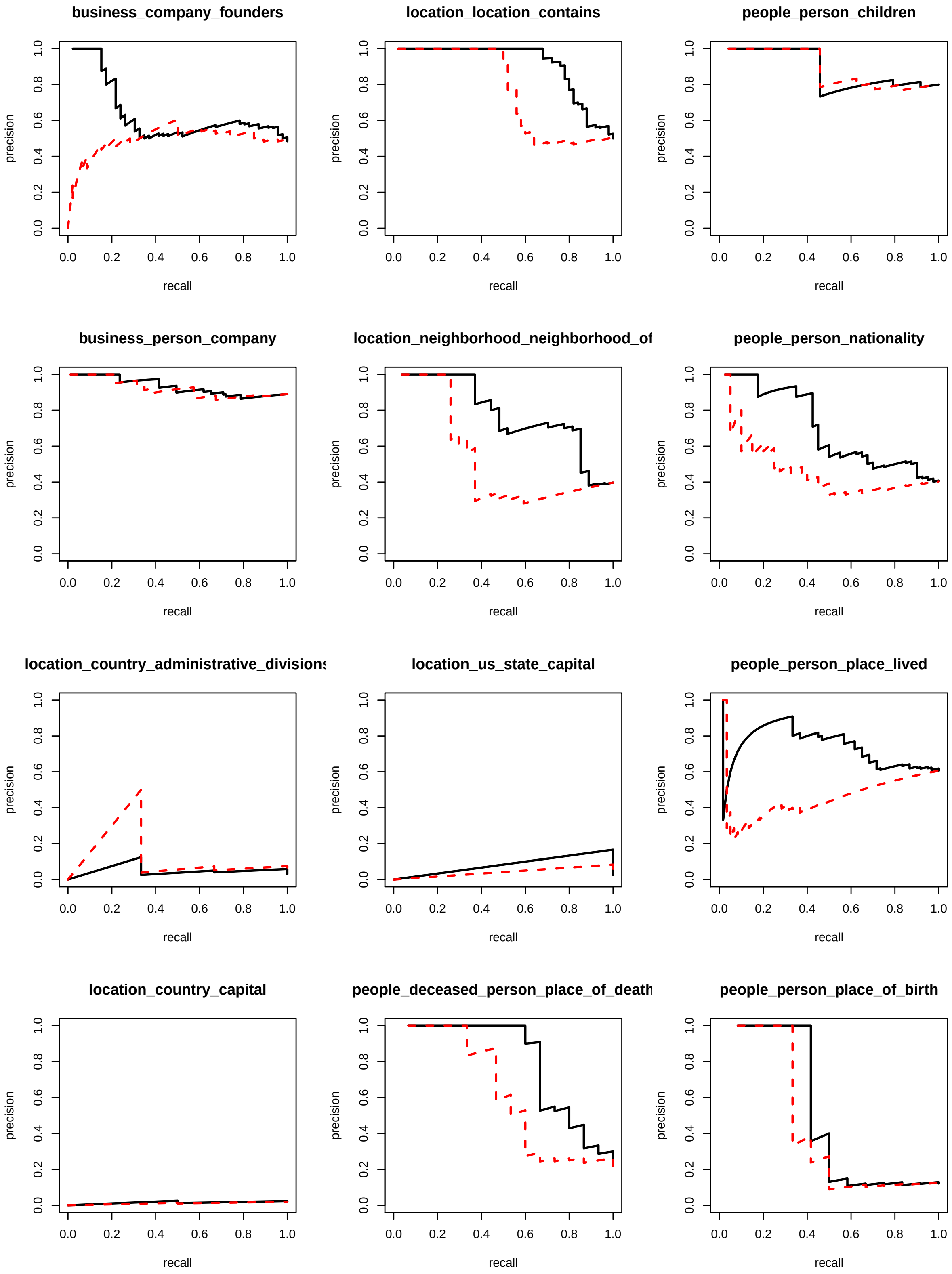


Figure 4.2: Sentence level PR curve

ing that missing data model is beneficial here and Algorithm 2 fails to perform well.

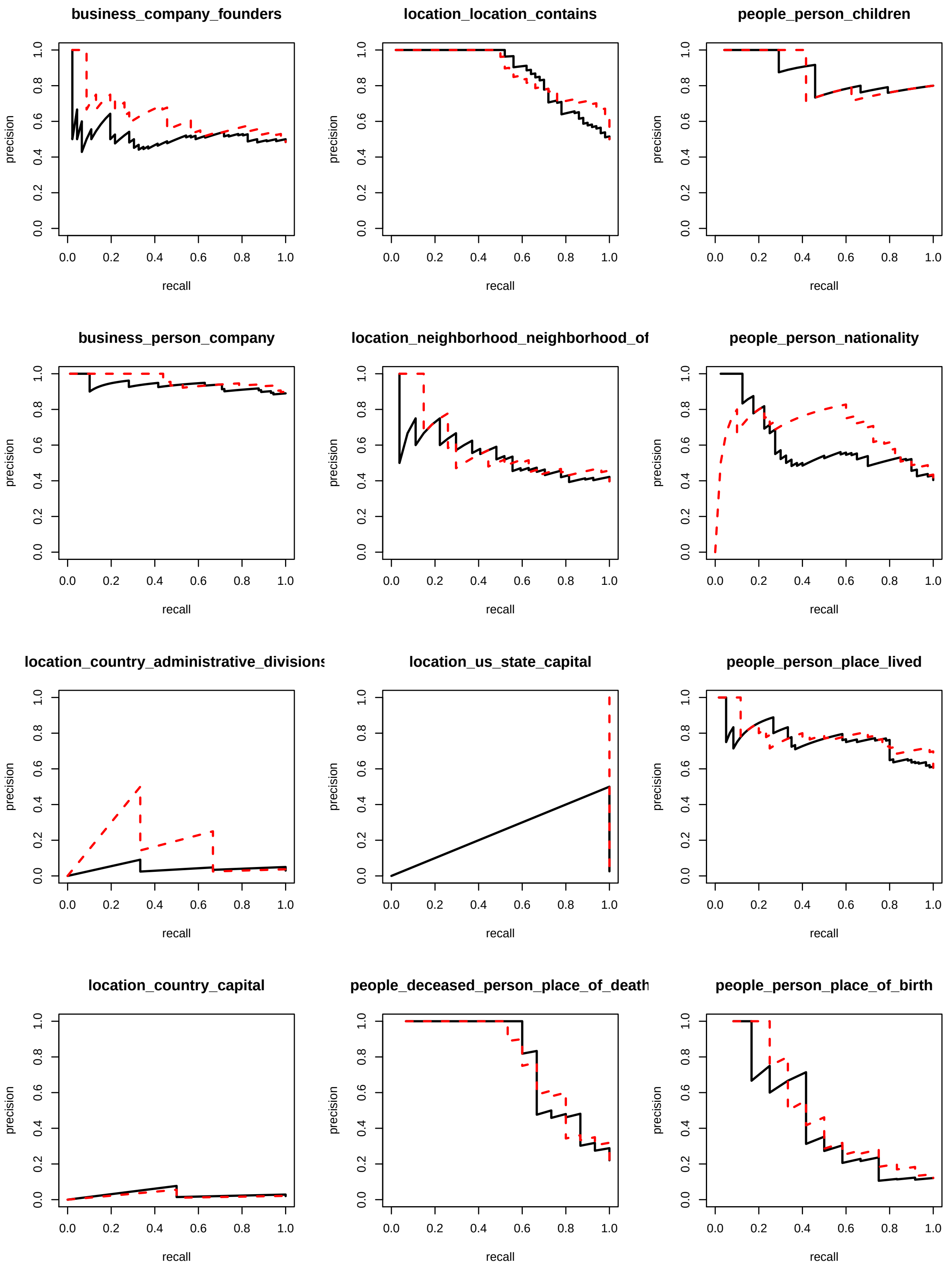


Figure 4.3: Sentence level PR curve with lexical features

Relation with lexical features	AUC for Algorithm - 1	AUC For Algorithm - 2
business/company/founder	0.49	0.61
location/location/contains	0.85	0.86
people/person/children	0.82	0.81
business/person/company	0.92	0.95
location/neighbourhood/ neighbourhood-of	0.50	0.54
people/person/nationality	0.58	0.66
location/country/ administrative-division	0.02	0.06
location/us state/capital	0.0	0.0
people/person/place-lived	0.74	0.77
location/country/capital	0.007	0.005
people/deceased-person/ place-of-death	0.7194	0.7191
people/person/place-of-birth	0.36	0.41

Table 4.2: Per-relation AUC of PR curve with Passage retrieval model

Figure 4.3 and Table 4.2 shows graphical and tabular representations of precision recall values and area under curve, respectively for the two algorithms. The dataset is preprocessed with two lexical feature - *Bag-of-words* and *Word-position*. This combined with pseudo-relevance relation feedback model of Algorithm 2 performs better than Algorithm 1. Here, again relations with more than 10% Freebase coverage like *location/location/contains* and *people/deceased-person/place-of-death* shows high AUC values. Relations like *location/country/administrative-division*, *location/us-state/capital* and *location/country/capital* due to less than 0.2% coverage has very low precision recall values.

Interestingly, relations *people/person/place-lived*, *business/person/company* and *location/neighbourhood/neighbourhood-of* shows high AUC values for both algorithms. These relations have relatively low coverage in Freebase (nearly 0.7%) but performs well showing that Algorithm 2 performs better when the data is pre-processed for lexical feature extraction model.

Chapter 5

Conclusions and Future Work

Our results give a comparative study of the two algorithms based on distant supervision. Both algorithms handle a crucial issue of missing data in the knowledge base with completely different approaches which have been neglected before, causing numerous false negatives.

Algorithm 1 introduces a latent variable model for learning and relaxes hard constraints providing a natural way of incorporating side information through a missing data model. To ensure efficient inferencing of a large dataset it introduces local search method with random and multiple starting points.

Algorithm 2 expands the knowledge base by first matching relation instances to sentences and learning the passage retrieval model and then providing the relevance feedback on sentences. These new relation instances are then added to the knowledge base and again the process is repeated to finally extract relations.

Analysing the results of performing relation extraction using distant supervision on two datasets, shows us that Algorithm 1 is more robust and efficient in extracting relations in scenarios where pre-processing of data is not possible. Such an algorithm is ideal for real-time relation extraction models where pre-processing is time consuming. However, in situations where more accurate result is required, without any time and resource constrain, dataset can be aligned with the passage retrieval model during pre-processing. Algorithm 2 has shown to perform better under such conditions, extracting successfully more relation entity pair in sentences compared to Algorithm 1. For some relations where instances both in text corpus and Freebase are few, both algorithms fail to obtain substantial information. Both algorithms perform well in particular conditions and an interesting direction to proceed will be to unify these two approaches of extracting relations in missing data model, such that we can achieve the performance of Algorithm 2 without pre-processing of data.

Distant supervision has been a topic of interest in the academia for some time now and so has been the problem of missing data. In future, we would like to apply these learning approaches with knowledge base to other tasks that could be modelled with semi-supervised learning algorithms, such as co-reference and name entities, etc.. Another interesting research area will be to include syntactic features along with the lexical features in distant supervision. Mintz et. al [MBSJ09] has performed experiments using only syntactic features for extraction but the combination of syntactic features and lexical features on missing data models will be a natural direction to proceed.

Bibliography

[AG00] Eugene Agichtein and Luis Gravano. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the fifth ACM conference on Digital libraries*, pages 85–94. ACM, 2000.

[BHB11] Edward Benson, Aria Haghighi, and Regina Barzilay. Event discovery in social media feeds. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 389–398. Association for Computational Linguistics, 2011.

[Bri99] Sergey Brin. Extracting patterns and relations from the world wide web. In *The World Wide Web and Databases*, pages 172–183. Springer, 1999.

[CBK+10] Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R Hruschka Jr, and Tom M Mitchell. Toward an architecture for never-ending language learning. In *AAAI*, volume 5, page 3, 2010.

[CK+99] Mark Craven, Johan Kumlien, et al. Constructing biological knowledge bases by extracting information from text sources. In *ISMB*, volume 1999, pages 77–86, 1999.

[Col02] Michael Collins. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 1–8. Association for Computational Linguistics, 2002.

[ECD+05] Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S Weld, and Alexander Yates. Unsupervised named-entity extraction from the web: An experimental study. *Artificial Intelligence*, 165(1):91–134, 2005.

[fre14] Freebase - online database, 2014.

[Hea92] Marti A Hearst. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics-Volume 2*, pages 539–545. Association for Computational Linguistics, 1992.

[HZL$^+$11] Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel 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: Human Language Technologies-Volume 1*, pages 541–550. Association for Computational Linguistics, 2011.

[KF09] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[LBCKT06] Percy Liang, Alexandre Bouchard-Côté, Dan Klein, and Ben Taskar. An end-to-end discriminative approach to machine translation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 761–768. Association for Computational Linguistics, 2006.

[MBSJ09] Mike Mintz, Steven Bills, Rion Snow, and Dan 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.

[MGW$^+$13] Bonan Min, Ralph Grishman, Li Wan, Chang Wang, and David Gondek. Distant supervision for relation extraction with an incomplete knowledge base. In *HLT-NAACL*, pages 777–782, 2013.

[Pop07] Ana-Maria Popescu. *Information extraction from unstructured web text*. PhD thesis, Citeseer, 2007.

[RYM10] Sebastian Riedel, Limin Yao, and Andrew McCallum. Modeling relations and their mentions without labeled text. In *Machine Learning and Knowledge Discovery in Databases*, pages 148–163. Springer, 2010.

[RZME13] A. Ritter, L. Zettlemoyer, Mausam, and O. Etzioni. Modelling missing data in distant supervision for information extraction. *TACL*, 2013.

[STNM12] Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher 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.

[TSN12] Shingo Takamatsu, Issei Sato, and Hiroshi 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.

[WW07] Fei Wu and Daniel S Weld. Autonomously semantifying wikipedia. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 41–50. ACM, 2007.

[XHZG13] W. Xu, R. Hoffmann, L. Zhao, and R. Grishman. Filling knowledge base gaps for distant supervision of relation extraction. *ACL*, 2013.

[ZC07] Luke S Zettlemoyer and Michael Collins. Online learning of relaxed ccg grammars for parsing to logical form. In *In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL-2007*. Citeseer, 2007.

Appendix

E.1 Per-relation occurrence in Dataset and Freebase

Relation	Fb	Train1	Test1	Train2	Test2
business/company/founder	1521	47	36	56	36
location/location/contains	853140	2066	819	2409	819
people/person/children	19675	49	22	53	22
business/person/company	35860	357	141	391	141
location/neighbourhood/neighbourhood-of	5939	139	38	311	38
people/person/nationality	601509	436	275	499	275
location/country/administrative-division	5081	59	66	59	66
location/us-state/capital	50	9	9	9	9
people/person/place-lived	16288	581	211	1083	211
location/country/capital	407	36	35	36	35
people/deceased-person/place-of-death	105758	190	46	521	46
people/person/place-of-birth	345023	370	113	644	113