A Comparison of Weak Supervision methods for Knowledge Base Construction

Ameet Soni
Department of Computer Science
Swarthmore College
soni@cs.swarthmore.edu

Dileep Viswanathan
School of Informatics and Computing
Indiana University
diviswan@indiana.edu

Niranjan Pachaiyappan
School of Informatics and Computing
Indiana University
nirapach@indiana.edu

Sriraam Natarajan
School of Informatics and Computing
Indiana University
natarasr@indiana.edu

Abstract

We present a comparison of *weak* and *distant* supervision methods for producing proxy examples for supervised relation extraction. We find that knowledge-based weak supervision tends to outperform popular distance supervision techniques, providing a higher yield of positive examples and more accurate models.

1 Introduction

In performing *relation extraction* in knowledge base population (KBP), the need for human-annotated examples (i.e., *gold-standard*) examples, is prohibitively expensive. One solution is to generate a set of so-called *silver-standard* examples from *weak* or *distant supervision* methods.

While several papers have demonstrated the benefits of using these approaches (Mintz et al., 2009; Riedel et al., 2010; Takamatsu et al., 2012), we are not familiar with any work that compares methods for generating weak labels for KBP. In this work, we seek to address the question of which weak supervision techniques provide the best basis for learning accurate models and scale appropriately with the KBP task. We address two approaches:

- Distant supervision (DS) – this popular technique entails referencing external knowledge bases, such as Freebase, as a source of seed facts. These facts are then linked to a corpus to identify positive training examples. We consider two variations for a corpus – extracting positive sentences from the actual training/testing corpus (CDS) (i.e., newswire documents) versus using sentences from external data sources (EDS) (e.g., Wikipedia articles).

- Knowledge-based weak supervision (KWS) – Natarajan et al. (2014) showed that we can encode the “world knowledge” of domain experts, who have some inherent rules for identifying positive training examples during manual annotation (e.g., “home teams are more likely to win a game” for a sports corpus). Using these rules, we can automatically generate new positive examples that simulate the human expert’s annotations in a training corpus.

In this paper, we present our approaches for generating examples in further detail. We evaluate all three approaches on the TAC KBP corpus. We will also describe our pipeline, which utilizes relational dependency networks (RDNs) (Neville and Jensen, 2007; Natarajan et al., 2010). We note that the central focus of this paper is not to showcase RDNs for this task – that has been done in previous work – but rather to investigate weak supervision techniques.

Our results show that knowledge-based weak supervision is the preferred choice for producing training examples when good rules are available, approaching the accuracy of gold-standard data sets. This method produces examples at a higher rate than DS with fewer mislabels and is flexible to adapt to a diverse set of relations. Distant supervision techniques scale quicker and excel when domain knowledge is difficult to encode. However, they tend to yield fewer results and are not applicable when a relevant database does not already exist.

Proceedings of AKBC 2016, pages 97–102,
San Diego, California, June 12-17, 2016. ©2016 Association for Computational Linguistics

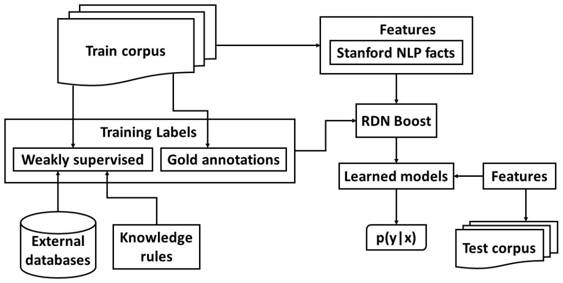

Figure 1: Full RDN relation extraction pipeline

2 The Relational Dependency Network Framework

In this work, we consider a framework for performing learning and inference from stochastic, noisy, relational data called Relational Dependency Networks (RDNs) (Neville and Jensen, 2007; Natarajan et al., 2010). RDNs extend dependency networks (DN) (Heckerman et al., 2001) to the relational setting. The key idea in a DN is to approximate the joint distribution over a set of random variables as a product of their marginal distributions, i.e., $P(y_1, ..., y_n|\mathbf{X}) \approx \prod_i P(y_i|\mathbf{X})$. It has been shown that employing Gibbs sampling in the presence of a large amount of data allows this approximation to be particularly effective. Note that, one does not have to explicitly check for acyclicity making these DNs particularly easy to be learned.

In an RDN, typically, each distribution is represented by a relational probability tree (RPT) (Neville et al., 2003). However, following previous work (Natarajan et al., 2010), we replace the RPT of each distribution with a set of relational regression trees (Blockeel and Raedt, 1998) built in a sequential manner i.e., replace a single tree with a set of gradient boosted trees. This approach has been shown to have state-of-the-art results in learning RDNs and we adapted boosting to learn for relation extraction. Since this method requires negative examples, we created negative examples by considering all possible combinations of entities that are not present in positive example set and sampled twice as many negatives as positive examples. We encourage the reader to refer to our previous work for in-depth details on the RDN algorithm (Neville et al., 2003; Natarajan et al., 2010).

The entire algorithmic pipeline is summarized in Figure 1. Given a training corpus of raw text documents, our learning algorithm first converts these documents into a set of facts (i.e., features) that are encoded in first order logic (FOL). Raw text is processed using the Stanford CoreNLP Toolkit[1] (Manning et al., 2014) to extract parts-of-speech, word lemmas, etc. as well as generate parse trees, dependency graphs and named-entity recognition information. The full set of extracted features is available in Table 1. These features are used to train our RDN model. For some (unlabeled) test corpus, the RDN model is utilized to perform inference and identify positive entities.

Feature	Description
wordString	word with word id
wordPosition	location of the word
caselessWordString	word string in lower case
wordLemma	canonical form of word
isNEWord	whether word is NE
nextWords	two succeeding words
prevWords	two preceding words
nextPOS	POS for the succeeding words
prevPOS	POS for the preceding words
nextLemmas	canonical form of successors
prevLemmas	canonical form of predecessors
nextNE	succeeding NE phrases
prevNE	preceding NE phrases
lemmaBetween	canonical form of word occurring between two NEs
neBetween	word b/w two NEs is an NE
posBetween	POS of word b/w two NEs
Dependency Path	
rootChildLemma	canonical form of child of DPR
rootChildNER	child of DPR is NE
rootChildPOS	POS of child of DPR
rootLemma	lemma of DPR
rootNER	DPR is NER
rootPOS	POS of DPR

Table 1: Features derived from the training corpus used by our learning system. POS - part of speech. NE - Named Entity. DPR - root of dependency path tree.

3 Weak Supervision Frameworks

We analyze two general types of weak supervision that have been successfully applied in other natural language tasks – distant supervision and knowledge-based weak supervision.

[1] http://stanfordnlp.github.io/CoreNLP/

3.1 Distant supervision

Distant supervision entails the use of external knowledge (e.g., a database) to heuristically label examples. Following standard procedure, we use three data sources – Never Ending Language Learner (NELL) (Carlson et al., 2010), Wikipedia Infoboxes and Freebase. For a given target relation, we identify relevant database(s), where the entries in the database form *entity pairs* (e.g., an entry of $(Barack\ Obama, Malia\ Obama)$ for a parent database) that will serve as a seed for positive training examples. These pairs must then be mapped to *mentions* in our corpus – that is, we must find sentences in our corpus that contain both entities together (Zhang et al., 2012). This process is done heuristically and is fraught with potential errors and noise (Riedel et al., 2010).

We identify two methods for mapping entities to mentions to create positive training examples. The first maps entity pairs to sentences in a training corpus native to our test domain (e.g., TAC KBP 2014). We refer to this as Corpus Distant Supervision, or **CDS**. This has the advantage of providing examples that are similar to the problem at hand and closer to the test queries. However, this can potentially omit thousands of example mentions that occur in a different context than the training corpus (e.g., a corpus of business articles will not contain matches to actors or sports players). As a result, most entity pairs in a database will fail to map to a corpus (while others may have several mappings).

To overcome this limitation, we alternatively map entity pairs to mentions in their corresponding Wikipedia article(s). By scraping these articles for relevant sentences, we hypothesize that we can detect a higher hit rate for each database entry. Any sentence containing the relevant entity pair is processed as a positive training example for our learning algorithm. We will refer to this technique as **EDS** for external-text distant supervision.

3.2 Knowledge-based weak supervision

While the literature supports distant supervision as a viable alternative, the quality of the generated labels is crucially dependent on the heuristic that is being used to map the relations to the knowledge base. As noted by Riedel et al. (2010), the distant supervision assumption can be too strong, particularly when the source used for labeling the examples is external to the learning task at hand.

Natarajan et al. (2014) proposed work based on the following insight: labels are typically created by "domain experts" who annotate the labels carefully, and who typically employ some inherent rules in their mind to create examples. For instance, consider identifying a person's family relationship from news articles. We may have an *inductive bias* towards believing two persons in a sentence with the same last name are related, or that the words "son" or "daughter" are strong indicators of a parent relation. We call this *world knowledge* as it describes the domain (or the world) of the target relation. We aim to use such knowledge to create examples for learning from text.

To this effect, we encode the domain expert's knowledge in the form of first-order logic rules with accompanying weights to indicate the expert's confidence. We use the probabilistic logic formalism *Markov Logic Networks* (Domingos and Lowd, 2009) to perform inference on unlabeled text (e.g., the TAC KBP corpus). Potential entity pairs from the corpus are queried to the MLN, yielding (weakly-supervised) positive examples. We choose MLNs as they permit domain experts to easily write rules while providing a probabilistic framework that can handle noise, uncertainty, and preferences while simultaneously ranking positive examples.

In our experiments, we found that the difference between multiple weight settings do not affect the results as long as the ordering between the rules is maintained – only the *scale* of the probabilities varies. We use the Tuffy system (Niu et al., 2011) to perform inference. The inference algorithm implemented inside Tuffy appears to be robust and scales well to millions of documents[2].

For the KBP task, some rules that we used are shown in Table 2. For example, the first rule identifies any number following a person's name and separated by a comma is likely to be the person's age (e.g., "Sharon, 42"). The third and fourth rule provide examples of rules that utilize more textual features; these rules state the appearance of the lemma

[2] As the structure and weights are pre-defined by the expert, learning is not needed for our MLN

Weight	MLN Clause
1.0	entityType(a, “PER”), entityType(b, “NUM”), nextWord(a, c), word(c, “,”), nextWord(c, b) → age(a, b)
0.6	entityType(a, “PER”), entityType(b, “NUM”), prevLemma(b, “age”) → age(a, b)
0.8	entityType(a, “PER”), entityType(b, “PER”), nextLemma(a, “mother”) → parents(a, b)
0.8	entityType(a, “PER”), entityType(b, “PER”), nextLemma(a, “father”) → parents(a, b)
0.6	entityType(a, “PER”), entityType(b, “PER”), lemmaBetween(a, b, ”husband”) → spouse(a, b)
1.0	entityType(a, ”ORG”), entityType(b, ”PER”), prevPrevLemma(b, ”found”), prevLemma(b, ”by”) → foundedBy(a, b)

Table 2: A sample of knowledge-based rules for weak supervision. The first value defines a weight, or confidence in the accuracy of the rule. The target relation appears at the end of each clause. “PER”, “ORG”, “NUM” represent entities that are persons, organizations, and numbers, respectively.

“mother” or “father” between two persons is indicative of a parent relationship (e.g.,“Malia’s father, Barack, introduced her...”).

4 Experiments and Results

We consider five TAC KBP relations from two categories, *person* and *organization*, chosen based on prior work for TAC KBP 2015. The relations are listed in the middle of Table 3 along with the counts of number of positives retrieved by each method (left). The TAC KBP 2014 corpus is used for training (or Wikipedia articles as is the case for EDS) while TAC KBP 2015 is used for testing. Another relation, age, is omitted from the results since a corresponding database is not available for distant supervision.

In our experiments, Freebase yields entity pairs for three out of the five relations ($siblings$, $spouse$, and $foundedBy$) while Wikipedia Infoboxes provides entity pairs for the $parents$ and $countryOfHeadquarters$ ($countryHQ$ for short) relations. NELL is utilized to supplement mentions for $siblings$. For KWS, a range of 4 to 8 rules are derived for each relation; only 5% of the training corpus was queried to generate KWS examples due – this proved sufficient for most relations although the number could easily be expanded as part of future work. Additionally, only the first 500 examples are actually utilized from Table 3. Performing larger runs is part of work in progress.

The results are obtained from averaging 5 different runs for each condition/relation. Across the runs, the test set is constant but the training set is subsampled with 75% membership to create more robust estimates of performance. The results are presented in right table of Table 3. We consider two standard metrics - area under the ROC curve and F1 score[3]. Table 3 also includes results after supplementing each weak/distant supervision with a small set (20) of gold-standard examples.

5 Discussion and Conclusions

Our experiments indicate strengths and weaknesses for each approach. The KWS framework, in general, outperforms the other methods in both metrics for the person relations, while being comparable for $countryHQ$. The tables do not even include age, the top performer for KWS. In fact, for several of the relations, the AUROC and F1 score approach or improve upon results when using a large, gold-standard training set (with the exception of $foundedBy$). For person relations, it produces more examples than either DS method, despite using only a fraction of the corpus. KWS struggles with creating good positive examples in our two organization relations, although it performed fairly similarly for headquarters despite limited examples. This was largely due to our inability to write discriminating rules that both achieves high recall and few false positives, demonstrating one drawback to the approach – the need for good (and formalisable) world knowledge.

Distant supervision did well where databases were easy to map – particularly, the organization relations and $parents$. There is no discernible difference between CDS and EDS, meaning that we can-

[3] Other metrics (e.g., accuracy, recall, AUPR) agree with the general conclusions presented here and are thus omitted

Positive Examples				AUROC			F1 score		
KWS	CDS	EDS	Relation	KWS	CDS	EDS	KWS	CDS	EDS
413	128	782	$parents$	**.68(.70)**	.62(.70)	.49(.67)	**.40(.40)**	.11(.19)	.08(.17)
1533	346	403	$spouse$	**.81(.83)**	.46(.49)	.53(.54)	**.24(.26)**	.04(.12)	.05(.08)
773	43	325	$siblings$	**.69(.73)**	.52(.64)	.58(.64)	**.26(.26)**	.10(.12)	.10(.21)
148	239	2207	$foundedBy$	.60(.65)	**.72(.74)**	.63(.67)	.21(.20)	**.26(.33)**	.26(.28)
21	168	1715	$countryHQ$	.58(.79)	.69(.72)	**.69(.80)**	.03(.57)	.06(.20)	.26(.43)

Table 3: LEFT Number of positive examples produced by each method per relation. The first three relations are person relations (per) while the last two describe organizations (org). **RIGHT** Area under the ROC curve (AUROC) and F1 measures for knowledge-based weak supervision (KWS), corpus distant supervision (CDS) and external source distant supervision (EDS). Numbers in parentheses are the results with 20 gold-standard examples. Emphasis indicates best performance for that relation.

not verify our hypothesis one way or the other (although the native corpus does slightly better in F1). This does give evidence that EDS is potentially useful when a large native corpus is not available for a task. If a large database exists with simple mapping heuristics, it can yield a large number of examples and perform well. Distant supervision is computationally faster and can easily scan an entire corpus, but MLNs yield a higher rate of positives and require less overhead.

Current efforts aim to expand upon these initial results by analyzing more relations (e.g., all 31 from TAC KBP) as well as by extending to other KB tasks such as medical abstract analysis. Furthermore, we were limited to only utilizing 5% of the corpus for KWS; an interesting question is whether there is a cap on the number of positives needed for good performance. Lastly, an interesting avenue of future work is whether the various weak supervision techniques can be combined together to achieve a more heterogeneous set of training examples. We hypothesize the data-centric approach of distant supervision would combine well with the knowledge-centric approach of KWS to achieve accuracies superior to even a large gold-standard set.

References

[Blockeel and Raedt1998] H. Blockeel and L. De Raedt. 1998. Top-down induction of first-order logical decision trees. *Artificial intelligence*, 101(1):285–297.

[Carlson et al.2010] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. Hruschka Jr., and T.Mitchell. 2010. Toward an architecture for never-ending language learning. In *Proceedings of the Twenty-Fourth Conference on Artificial Intelligence (AAAI)*.

[Domingos and Lowd2009] P. Domingos and D. Lowd. 2009. *Markov Logic: An Interface Layer for AI*. Morgan & Claypool, San Rafael, CA.

[Heckerman et al.2001] D. Heckerman, D. Chickering, C. Meek, R. Rounthwaite, and C. Kadie. 2001. Dependency networks for inference, collaborative filtering, and data visualization. *Journal of Machine Learning Research*, pages 49–75.

[Manning et al.2014] C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, and D. McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.

[Mintz et al.2009] M. Mintz, S. Bills, R. Snow, and D. Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*.

[Natarajan et al.2010] S. Natarajan, T. Khot, K. Kersting, B. Gutmann, and J. Shavlik. 2010. Boosting relational dependency networks. In *Proceedings of the International Conference on Inductive Logic Programming (ILP)*.

[Natarajan et al.2014] S. Natarajan, J. Picado, T. Khot, K. Kersting, C. Re, and J. Shavlik. 2014. Effectively creating weakly labeled training examples via approximate domain knowledge. In *International Conference on Inductive Logic Programming*.

[Neville and Jensen2007] J. Neville and D. Jensen. 2007. Relational dependency networks. In *Introduction to Statistical Relational Learning*. MIT Press.

[Neville et al.2003] J. Neville, D. Jensen, L. Friedland, and M. Hay. 2003. Learning relational probability trees. In *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 625–630.

[Niu et al.2011] F. Niu, C. Ré, A. Doan, and J. W. Shavlik. 2011. Tuffy: Scaling up statistical inference in Markov logic networks using an RDBMS. *Proceedings of Very Large Data Bases (PVLDB)*, 4(6):373–384.

[Riedel et al.2010] S. Riedel, L. Yao, and A. McCallum. 2010. Modeling relations and their mentions without labeled text. In *Proceedings of the 2010 European conference on Machine learning and knowledge discovery in databases (ECML KDD)*.

[Takamatsu et al.2012] S. Takamatsu, I. Sato, and H. Nakagawa. 2012. Reducing wrong labels in distant supervision for relation extraction. In *Proceedings of the Association for Computational Linguistics (ACL)*.

[Zhang et al.2012] C. Zhang, F. Niu, C. Ré, and J. Shavlik. 2012. Big data versus the crowd: Looking for relationships in all the right places. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, pages 825–834.