To appear in Proceedings of Inductive Logic Programming (ILP) 2014

# Effectively creating weakly labeled training examples via approximate domain knowledge

Sriraam Natarajan, Jose Picado[#], Tushar Khot[*], Kristian Kersting[+],
Christopher Re[**], Jude Shavlik[*]
Indiana University, USA, [#] Oregon State University, USA
[*] University of Wisconsin-Madison, USA,[**] Stanford University, USA
[+] Technical University of Dortmund, Germany

**Abstract.** One of the challenges to information extraction is the requirement of human annotated examples, commonly called gold-standard examples. Many successful approaches alleviate this problem by employing some form of distant supervision, i.e., look into knowledge bases such as Freebase as a source of supervision to create more examples. While this is perfectly reasonable, most distant supervision methods rely on a hand-coded background knowledge that explicitly looks for patterns in text. For example, they assume all sentences containing Person X and Person Y are positive examples of the relation *married(X, Y)*. In this work, we take a different approach – we infer weakly supervised examples for relations from models learned by using knowledge outside the natural language task. We argue that this method creates more robust examples that are particularly useful when learning the entire information-extraction model (the structure and parameters). We demonstrate on three domains that this form of weak supervision yields superior results when learning structure compared to using distant supervision labels or a smaller set of gold-standard labels.

## 1 Introduction

Supervised learning is one of the popular approaches to information extraction from natural language (NL) text where the goal is to learn relationships between attributes of interest – learn the individuals employed by a particular organization, identifying the winners and losers in a game, etc. There have been two popular forms of supervised learning used for information extraction. First is the classical machine learning approach. For instance, the NIST Automatic Content Extraction (ACE) RDC 2003 and 2004 corpora, has over 1000 documents that have human-labeled relations leading to over $16,000$ relations in the documents [12]. ACE systems use textual features – lexical, syntactic and semantic – to learn mentions of target relations [27, 21]. But pure supervised approaches are quite limited in scalability due to the requirement of high quality labels, which can be very expensive to obtain for most NL tasks. An attractive second approach is *distant supervision*, where labels of relations in the text are created by applying a heuristic to a common knowledge base such as Freebase [12, 19, 23]. The quality of these labels are crucially dependent on the heuristic used to map the relations to the knowledge base. Consequently, there have been several approaches that aim to improve the quality of these labels ranging from multi-instance learning [19, 5, 22] to using patterns that frequently appear in the text [23]. As noted by Riedel et al. [19], 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.

Hence, we use the probabilistic logic formalism called *Markov Logic Networks* [4] to perform *weak supervision* [1] to create more examples. Instead of directly obtaining the labels from a different source, we perform inference on *outside knowledge* (i.e., knowledge not explicitly stated in the corpus) to create sets of entities that are "potential" relations. This outside knowledge forms the *context MLN* – CMLN – to reflect that they are non-linguistic models. An example of such knowledge could be that "home team are more likely to win a game". Note that this approach enables the domain expert to write rules in first-order logic so that the knowledge is not specific to any particular textural wording but is general knowledge about the world (in our example, about the games played). During the information extraction (IE) phase, unlabeled text are then parsed through some entity resolution parser to identify potential entities. These entities are then used as queries to the CMLN which uses data from non-NLP sources to infer the posterior probability of relations between these entities. These inferred relations become the probabilistic examples for IE. This is in contrast to distant supervision where statistical learning is employed at "system build time" to construct a function from training examples.

Our hypothesis – which we verify empirically – is that the use of world knowledge will help in learning from NL text. This is particularly true when there is a need to learn a model without any prior structure (e.g. a MLN) since the number of examples needed can be large. These weakly supervised examples can augment the gold-standard examples to improve the quality of the learned models. So far, the major hurdle to learning structure in IE is the large number of features leading to increased complexity in the search [17]. Most methods use a prior designed graphical model and **only** *learn the parameters*. A key issue with most structure-learning methods is that, when scoring every candidate structure, parameter learning has to be performed in the inner loop. We, on the other hand, employ an algorithm based on *Relational Functional Gradient Boosting* (RFGB) [13, 9, 8] for learning **the structure**. It must be emphasized clearly that the main contribution of the paper is not the learning algorithm, but instead is presenting a method for generation of weakly supervised examples to augment the gold standard examples.

We then employ RFGB in three different tasks:

1. Learning to jointly predict game winners and losers from NFL news articles[1]. We learned from $50$ labeled documents and used $400$ unlabeled documents. For the unlabeled documents, we used a common publicly available knowledge base such as Freebase to perform inference on the game winners and losers
2. Classifying documents as either *football* or *soccer* articles (using the American English senses of these words) from a data set that we created
3. Comparing our weak supervision approach to the distant supervision approach on a standard NYT data set[2].

[1] LDC catalog number LDC2009E112

[2] LDC catalog number LDC2008T19

We perform 5-fold cross validation on these tasks and show that the proposed approach outperforms learning only from gold-standard data.

When we can bias the learner with examples created from commonsense knowledge, we can distantly learn model structure. Because we have more training examples than the limited supply of gold-standard examples and they are of a higher quality than traditional distant labeling, the proposed approach allows for a better model to be learned. Our algorithm has two phases:

1 *Weak supervision phase*, where the goal is to use commonsense knowledge (CMLN). This CMLN could contain clauses such as “Higher ranked teams are more likely to win”.

2 *Information extraction phase*, where the noisy examples are combined with some “gold-standard” examples and a relational model is learned using RFGB on textual features.

The potential of such advice giving method is not restricted to NL tasks and is more broadly applicable. For instance, this type advice can be used for labeling tasks [24] or to shape rewards in reinforcement learning [2] or to improve the number of examples in a medical task. Such advice can also be used to provide guidance to a learner in unforseen situations [11].

## 2 Related work

*Distant Supervision:* Our approach is quite similar to the *distant supervision* [1, 12] that generate training examples based on external knowledge bases. Sentences in which any of the related entities are mentioned, are considered to be positive training examples. These examples along with the few annotated examples are provided to the learning algorithm. These approaches assume that the sentences that mention the related entities probably express the given relation. Riedel et al. [19] relax this assumption by introducing a latent variable for each mention pair to indicate whether the relation is mentioned or not. This work was further extended to allow overlapping relations between the same pair of entities (e.g. `Founded(Jobs, Apple)` and `CEO-of(Jobs, Apple)`) by using a multi-instance multi-label approach [5, 22]. We employ a model based on non-linguistic knowledge to generate the distant supervision examples. Although we rely on a knowledge base to obtain the relevant input relations for our CMLN model, one can imagine tasks where such relations are available as inputs or extracted earlier in the pipeline.

*Statistical Relational Learning:* Most NLP approaches define a set of features by converting structured output such as parse trees, dependency graphs, etc. to a flat feature vector and use propositional methods such as logistic regression. Recently, there has been a focus of employing Statistical Relational models that combine the expressiveness of first-order logic and the ability of probability theory to model uncertainty. Many tasks such as BioNLP [10] and TempEval [25] have been addressed [18, 16, 26] using SRL models, namely Markov Logic Networks (MLNs) [4]. But these approaches still relied on generating features from structured data. Sorower et al. [20] use a similar concept in spirit where they introduce a mention mode that models the probability of facts mentioned in the text and use a EM algorithm to learn MLN rules to achieve this. We represent the structured data (e.g. parse trees) using first-order logic and use the RFGB algorithm to learn the structure of Relational Dependency Networks (RDN) [14]. Relational Dependency Networks (RDNs) are SRL models that consider a joint distribution as a product of conditional distributions. One of the important advantages of RDNs is that the models are allowed to be cyclic. As shown in the next section, we use MLNs to specify the weakly supervised world knowledge.

# 3 Structure Learning for Information Extraction Using Weak Supervision

One of the most important challenges facing many natural language tasks is the paucity of "gold standard" examples. Our proposed method, shown in Figure 1, has two distinct phases: *weak supervision phase* where we create weakly supervised examples based on commonsense knowledge and *information extraction phase* where we learn the structure and parameters of the models that predict relations using textual features.

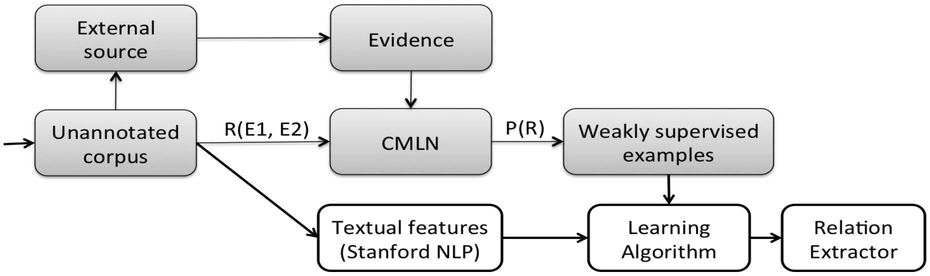

**Fig. 1.** Flowchart of our method. The top-half represents the weak supervision phase where we generate the examples using the CMLN and facts from an external source. The bottom-half represents the information extraction phase where we learn a SRL model using the weakly supervised and gold standard examples.

## 3.1 Weak Supervision Phase

We now explain how our first phase addresses the key challenge of obtaining additional training examples. As mentioned earlier, the key challenge is obtaining annotated examples. To address this problem, we employ a method that is commonly taken by humans. For instance, consider reading a newspaper sports section about a particular sport (say the NFL). We have an inherent *inductive bias* – we expect a high ranked team (particularly if it plays at home) to win. In other words, we rarely expect "upsets". We aim to formalize this notion by employing a model that captures this inductive bias to label in addition to gold standard examples.

We employ MLNs to capture this world knowledge. MLNs  [4] are relational undirected models where first-order logic formula correspond to the cliques of a Markov network and formula weights correspond to the clique potentials. A MLN can be instantiated as a Markov network with a node for each ground predicate (atom) and a clique for each ground formula. All groundings of the same formula are assigned the same weight. So the joint probability distribution over all atoms is

$$P(X \; = \; x) = \frac{1}{Z} \exp\left(\sum_i w_i n_i(x)\right) \tag{1}$$

where $n_i(x)$ is the number of times the $i$th formula is satisfied by possible world $x$ and $Z$ is a normalization constant. Intuitively, a possible world where formula $f_i$ is true one more time than a different possible world is $e^{w_i}$ times as probable, all other things being equal. There have been several weight learning, structure learning and inference algorithms proposed for MLNs. MLNs provide an easy way for domain's expert to specify the background knowledge and effective algorithms exist for learning the weights of these clauses and perform inference. We use the scalable Tuffy system [15] to perform inference. One of the key attractions of Tuffy is that it can scale to millions of documents .

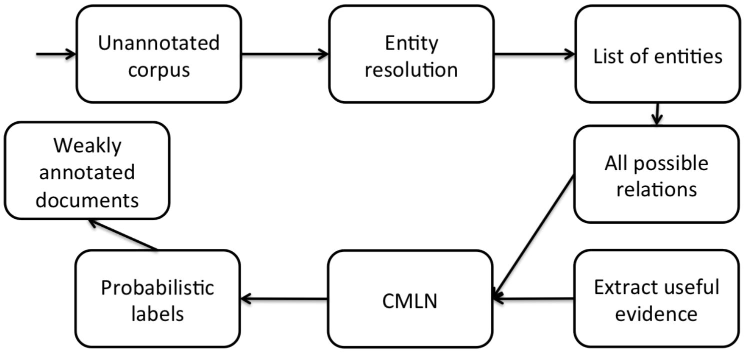

**Fig. 2.** Steps involved in creation of weakly supervised examples.

Our proposed approach for weak supervision is presented in Figure 2. The first step is to design a MLN that captures domain knowledge, called as *CMLN*. For the NFL domain, some rules that we used are shown in Table 1. For example, “Home team is more likely to win the game” (first two clauses), “High ranked team is more likely to win the game” (last two rules). Note that the rules are written without having the knowledge base in mind. These rules are simply written by the domain's expert and they are softened using a knowledge base such as Wikipedia. The resulting weights are presented in the left column of the table. We used the games played in the last 20 years to compute these weights.

While it is possible to learn these weights from data (for instance using previously played NFL games), we set the weights based on the log-odds for CMLN. For instance, for NFL domain, we set the weights of the clauses by considering the previously played NFL games[3]. We looked at the number of games played, the total number of times a home team won and the total number of times a higher ranked[4] team won, etc. If a home team won 10 times roughly more compared to away team, the weights were set to be $log(10) = 1$ for the rule about home team winning more often. Using this data, we set the weights of the MLN clauses as shown in Table 1. This is an approach that has been taken earlier when employing the use of MLNs [7, 6]. In our experiments, we found that the results are not very sensitive to the exact MLN weights as long as the order of the rule weights is preserved.

Note that one could simply define a higher ranking using the following MLN clause where $t$ denotes a team, $r$ its rank, $y$ the year of the ranking and $hR$ the higher rank: $\infty \quad rank(t1, r1, y), rank(t2, r2, y), t1! = t2, r1 < r2 \rightarrow hR(t1, t2, y)$. We argue that this is one of the major features of using common-sense knowledge. The relative merits between the different rules can be judged reasonably even though a domain's expert may not fully understand the exact impact of the weights. It is quite natural in several tasks, as we show empirically, to set "reasonable" weights.

| | |
|---|---|
| 0.33 | home(g, t) → winner(g, t) |
| 0.33 | away(g, t) → loser(g, t) |
| $\infty$ | exist t2 winner(g, t1), t1 != t2 → loser(g, t2) |
| $\infty$ | exist t2 loser(g, t1), t1 != t2 → winner(g, t2) |
| 0.27 | tInG(g, t1), tInG(g, t2), hR(t1, t2, y) → winner(g, t1) |
| 0.27 | tInG(g, t1), tInG(g, t2), hR(t1, t2, y) → loser(g, t2) |

**Table 1.** A sample of CMLN clauses used for the NFL task with their corresponding weights on the left column. $t$ denotes a team, $g$ denotes a game, $y$ denotes the year, $tInG$ denotes that the team $t$ plays in game $g$, $hR(t1, t2, y)$ denotes that $t1$ is ranked higher than $t2$ in year $y$. A weight of $\infty$ means that the clause is a "hard" constraint on the set of possible worlds, a weight different from $\infty$ means that the clause is a "soft" contraint.

The next step is to create weakly supervised learning examples. We identify interesting (unannotated) documents – for example, sport articles from different news web sites. We use *Stanford NLP* toolkit to perform entity resolution to identify the potential teams, games and year in the document. Then, we use the CMLN to obtain the posterior probability on the relations being true between entities mentioned in the same sentence – for example, game winner and loser relations. Note that to perform inference, evidence is required. Hence, we use the games that have been played between the two teams (again from previously played games that year) to identify the home, away and ranking of the teams. We used the rankings at the start of the year of the game as a pseudo reflection of the relative rankings between the teams.

Recall that the results of inference are the posterior probabilities of the relations being true between the entities extracted from the same sentence and they are used

[3] We obtained from Pro-Football-Reference http://www.pro-football-reference.com/

[4] According to http://www.nfl.com/

for annotations. One simple annotation scheme is using the MAP estimate (i.e., if the probability of a team being a winner is greater than the probability of being the loser, the relation becomes a positive example for winner and a negative example for loser). An alternative would be to use a method that directly learns from probabilistic labels. Choosing the MAP would make a strong commitment about several examples on the borderline. Since our world knowledge is independent of the text, it may be the case that for some examples perfect labeling is not easy. In such cases, using a softer labeling method might be more beneficial. Now these weakly supervised examples are ready for our next step – *information extraction*.

### 3.2 Learning for Information Extraction

Once the weakly supervised examples are created, the next step is inducing the relations. We employ the procedure from Figure 3. We run both the gold standard and weakly supervised annotated documents through Stanford NLP toolkit to create linguistic features. Once these features are created, we run the RFGB algorithm [13]. This allows us to create a joint model between the target relations, for example, game winner and losers. We now briefly describe the adaptation of RFGB to this task.

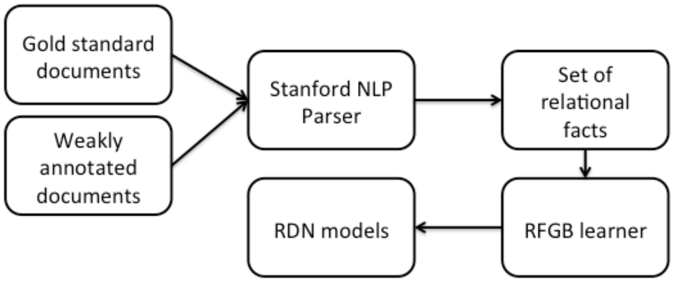

**Fig. 3.** Steps involved in learning using probabilistic examples.

Let the training examples be of the form $(\mathbf{x}_i, y_i)$ for $i = 1, ..., N$ and $y_i \in \{1, ..., K\}$. $\mathbf{x}$ denotes the features which in our case are lexical and syntactic features and $y$s correspond to target (game winners and loser) relations. Relational models tend to consider training instances as "mega examples" where each example represents all instances of a particular group. We consider each document to be a mega example i.e., we do not consider cross document learning.

The goal is to fit a model $P(y|\mathbf{x}) \propto e^{\psi(y,\mathbf{x})}$ for every target relation $y$. Functional gradient ascent starts with an initial potential $\psi_0$ and iteratively adds gradients $\Delta_i$. $\Delta_m$ is the functional gradient at episode $m$ and is

$$\Delta_m = \eta_m \times E_{x,y}[\partial/\partial\psi_{m-1} log\, P(y|x; \psi_{m-1})] \quad (2)$$

where $\eta_m$ is the learning rate. Dietterich *et al.* [3] suggested evaluating the gradient for every training example and fitting a regression tree to these derived examples $([(x_i, y_i), \Delta_m(y_i; x_i)])$.

In our formalism, $y$ corresponds to target relations, for example $gameWinner$ and $gameLoser$ relations between a team and game mentioned in a sentence. $\mathbf{x}$ corresponds to all the relational facts produced by Stanford NLP toolkit – lexical features, such as base forms of words, part-of-speech tags, word lemmas and entity types, and syntactic features such as phrase chunks, phrase types, parse trees and dependency paths. To learn the model for a relation, say $gameWinner$, we start with an initial model $\psi_0$ (uniform distribution). Next, we calculate the gradients for each example as the difference between the true label and current predicted probability. Then, we learn a relational regression tree to fit the regression examples and add it to the current model. We now compute the gradients based on the updated model and repeat the process. In every subsequent iteration, we *fix* the errors made by the model. For further details, we refer to Natarajan et al. [13].

Since we use a probabilistic model to generate the weakly supervised examples, our training examples will have probabilities associated with them based on the predictions from CMLN. We extend RFGB to handle probabilistic examples by defining the loss function as the KL-divergence[5] between the observed probabilities (shown using $P_{obs}$) and predicted probabilities (shown using $P_{pred}$). The gradients for this loss function is the difference between the observed and predicted probabilities.

$$\begin{aligned}\Delta_m(x) &= \frac{\partial}{\partial \psi_{m-1}} \sum_{\hat{y}} P_{obs}(y = \hat{y}) \log \left( \frac{P_{obs}(y = \hat{y})}{P_{pred}(y = \hat{y} | \psi_{m-1})} \right) \\ &= P_{obs}(y = 1) - P_{pred}(y = 1 | \psi_{m-1})\end{aligned}$$

Hence the key idea in our work is to use probabilistic examples that we obtain from the weakly supervised phase as input to our structure learning phase along with gold standard examples and their associated documents. Then a RDN is induced by learning to predict the target relations jointly, using features created by the Stanford NLP toolkit. Since we are learning a RDN, we do not have to explicitly check for acyclicity. We chose to employ RDNs as they have been demonstrated to have the state-of-the-art performance in many relational and noisy domains [13]. We use modified ordered Gibbs sampler [14] for inference.

## 4 Experimental results

We now present the results of empirically validating our approach on three domains. In the first two domains, we compare the use of augmenting with weakly supervised examples against simply using the gold standard examples. In the third domain, we compare the examples generated using our weak supervision approach against distant supervision examples on a standard data set. We also compare against one more distant supervision method that is the state-of-the-art on this data set.

[5] $D_{KL}(P;Q) = \sum_y P(y) log(P(y)/Q(y))$

### 4.1 NFL Relation Extraction

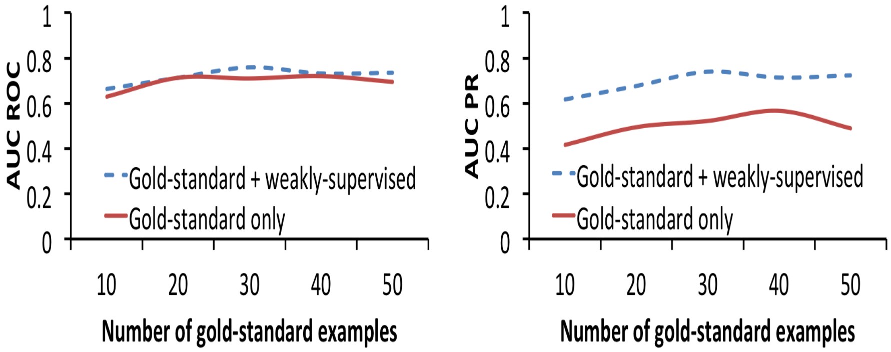

**Fig. 4.** Results of predicting winners and losers in NFL : **(a)** AUC ROC. **(b)** AUC PR.

The first data set on which we evaluate our method is the National Football League (NFL) data set from LDC[6] that consists of articles of NFL games from past two decades. The goal is to identify relations such as *winner* and *loser*. For example, consider the text, "Packers defeated Cowboys $28 - 14$ in Saturday's Superbowl game". The goal is to identify *Greenbay* and *Dallas* as the winner and loser respectively. The corpus consists of articles, some of which are annotated with target relations. We consider only articles that have annotations of positive examples. There were $66$ annotations of the relations. We used $16$ of these annotations as the test set and performed training on the rest. In addition to the gold standard examples, we used articles from the NFL website[7] for weak supervision. We used the MLN presented in Table 1 for inferring the weakly supervised examples.

The goal is to evaluate the impact of the weakly supervised examples. We used $400$ weakly supervised examples as the results did not improve beyond using $400$ examples. We varied the number of gold standard examples while keeping the number of weakly supervised examples constant. We compared against using only gold standard examples. The results were averaged over $5$ runs of random selection of gold standard examples. We measured the area under curves for both ROC and PR curves. Simply measuring the accuracy on the test set will not suffice as predicting the majority class can lead in high performance. Hence we present AUC. The results are presented in Figure 4 where the performance measure is presented by varying the number of gold standard examples. As can be seen, in both metrics, the weakly supervised examples improve upon the usage of gold standard examples. The use of weakly supervised examples allows a more accurate performance with a small number of examples, a steeper

[6] http://www.ldc.upenn.edu
[7] http://www.nfl.com

learning curve and in the case of PR, convergence to a higher value. It should be mentioned that for every point in the graph, we sample the gold standard examples from a fixed set of examples and the only difference is whether there are any weakly supervised examples added. For example, when plotting the results of $10$ examples, the set of gold standard examples is the same for every run. For the blue dashed curve, we add $400$ more weakly supervised examples and this is repeated for $5$ runs in which the $10$ gold examples are drawn randomly.

We also performed t-tests on all the points of the PR and ROC curves. For the PR curves, the use of weakly supervised learning yields statistically superior performance over the gold standard examples for all the points on the curves (with p-value $< 0.05$). For the ROC curves, significance occurs when using $10$ and $30$ examples. Since PR curves are more conservative than ROC curves, it is clear that the use of these weakly supervised examples improves the performance of the structure learner significantly. To understand whether weak supervision helps, we randomly assigned labels to the $400$ examples (instead of using weak supervision). When combined with $50$ gold examples, the performance decreased dramatically with AUC values being less than $0.6$. This clearly shows that the weakly supervised labels help when learning the structure. We also used the MAP estimates from the MLN to label the weakly supervised examples. The performance was comparable to that of using random labels labels across all runs with an average AUC-ROC of $0.6$.

### 4.2 Document Classification

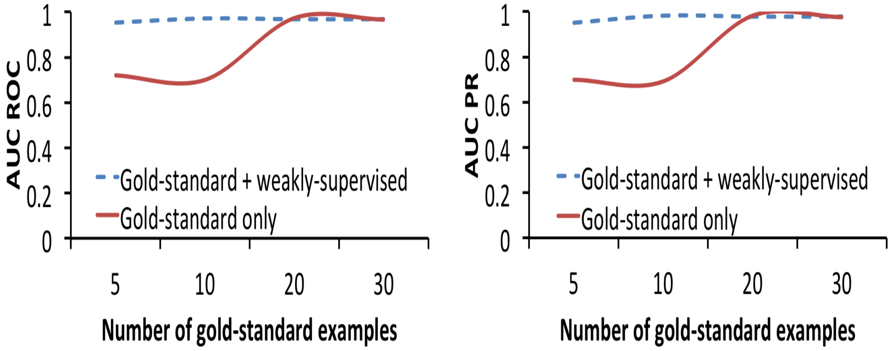

**Fig. 5.** Results of classifying football and soccer articles in the Document Classification domain: **(Left)** AUC ROC. **(Right)** AUC PR.

As a second experiment, we created another data set where the goal is to classify documents either as being *football(American)* or *soccer* articles. The target relation is the type of the article. We extracted $30$ football articles from the NFL website[8] and $30$

[8] http://www.nfl.com

soccer articles from the English Premier League (EPL) website[9] and annotated them manually as being football and soccer respectively. We used only the first paragraph of the articles for learning the models since it appeared that enough information is present in the first paragraph for learning an useful model. We purposefully annotated a small number of articles as the goal is to analyze performance under a small number of gold standard examples. In addition, we used 45 articles for weak supervision (as with the previous case, the performance did not improve with more weakly supervised articles). We used rules such as, "If a soccer league and a soccer team are mentioned, then it is a soccer game", "If a football league and a football team are mentioned, then it is a football game", "EPL teams play soccer", "NFL teams play football", "If the scores of both teams are equal or greater than 10, then it is a football game", "If the scores of both teams are less than 10, then it is a soccer game", "If the scores of both teams are 0, then it is a soccer game". These are presented in Table 2.

All the rules mentioned are essentially considered as "soft" rules. The weights of these rules were simply set to 100, 10, 1 to reflect the log-odds. During the weak supervision phase, we used the entities mentioned in the documents as queries to CMLN to predict the type. These predictions (probabilities) become the weak supervision for the learning phase. As with NFL, we measured the AUC ROC and PR values by varying the number of gold standard examples. Again, in each run, to maintain consistency, we held the gold standard examples to be constant and simply added the weakly supervised examples. The results are presented in Figure 5. The resulting figures show that as with the earlier case, weak supervision helps improve the performance of the learning algorithm. We get a jump start and a steeper learning curve in this case as well. Again, the results are statistically significant for small number of gold standard examples. The experiment proves that adding probabilistic examples as weak supervision improves performance. Note that with more gold standard examples, the performance decreases due to overfitting. But with weak supervision, since the examples are probabilistic, this issue is avoided – another observation that the use of weaker examples aids the learner.

| | |
|---|---|
| 100 | te(a,t), le(a,l), tInL(t,l), sL(l) → soccer(a) |
| 100 | te(a,t), le(a,l), tInL(t,l), fL(l) → football(a) |
| 10 | te(a,t), sL(l), tInL(t,l) → soccer(a) |
| 10 | te(a,t), fL(l), tInL(t,l) → football(a) |
| 1 | te(a,t1), te(a,t2), sco(a,t1,s1), sco(a,t2,s2), s1 $>=$ 10, s2 $>=$ 10 → football(a) |
| 1 | te(a,t1), te(a,t2), sco(a,t1,s1), sco(a,t2,s2), s1 $<$ 10, s2 $<$ 10 → soccer(a) |
| 1 | te(a,t1), te(a,t2), sco(a,t1,s1), sco(a,t2,s2), s1 $=$ 0, s2 $=$ 0 → soccer(a) |

**Table 2.** CMLN clauses used for document classification. $a$ denotes an article, $t$ denotes a team, $l$ denotes a league, $te$ denotes that $t$ is a team mentioned in article $a$, $le$ denotes that $l$ is a league mentioned in article $a$, $tInL$ denotes that the team $t$ plays in league $l$, $sL$ denotes that $l$ is a soccer league, $fL$ denotes that $l$ is a football league, $sco$ denotes that $t$ had a final score of $s$ in $a$.

[9] http://www.premierleague.com

### 4.3 NY Times Relation Extraction

The question that we explicitly aim to ask in this experiment is: How does this method of weak supervision compare against a distant supervision method to create the examples for structure-learning? We used the corpus created by Riedel et al. [19] by aligning the NYT corpus between the years 2005 and 2007 with Freebase[10] relations. The same corpus was subsequently used by Hoffmann et al. [5] and Surdeanu et al. [22]. We restricted our attention to only the *nationality* relation. We performed $5$ runs with $70$ examples for training and $50$ examples for testing. Surdeanu et al. [22] exhibit the state-of-the-art performance in this task. But they **only learn the parameters** of the graphical model while we learn the entire model. Hence we did not compare against their method.

To generate probabilistic examples, we designed an MLN and set its weights to predict the nationality of a person based on his or her place of birth and location of residence. The MLN is presented in Table 3. We queried Freebase for the place of birth of a person and the places the person has lived in, and used this information as evidence. We used rules such as "If a person was born in a country, then the person's nationality is that country" and "If a person has lived in a country, then the person's nationality is that country" (first and third clauses). Because the place of birth or location of residence provided by Freebase are not always countries, we also queried Freebase for the countries in which the locations are contained. We also used rules rules such as "If a person was born in a location (e.g., a city) and that location is contained in a country, then the person's nationality is that country" and "If a person has lived in a location and that location is contained in a country, then the person's nationality is that country" (second and fourth clauses).

We queried the MLN to obtain the posterior probabilities on the *nationality* relation between each person and country. Finally we used the person and country entities, as well as the corresponding posterior probabilities, to create new probabilistic examples for RFGB.

| | |
|---|---|
| 1 | place_of_birth(p, c) → nationality(p, c) |
| 1 | place_of_birth(p, l), contained_by(l, c) → nationality(p, c) |
| 0.1 | place_lived(p, c) → nationality(p, c) |
| 0.1 | place_lived(p, l), contained_by(l, c)→ nationality(p, c) |

**Table 3.** CMLN used for the NYT relation extraction. $p$ denotes a person, $l$ denotes a location, $c$ denotes a country.

We compare our examples to the distant supervision examples obtained from Freebase. To obtain the distant supervision examples, we queried Freebase for the nationality of a person and looked for sentences in the dataset that contained each pair of entities. We used the same structure-learning approach for the weak supervision (obtained from the MLN) and distant supervision (obtained from Freebase) examples. The results comparing both settings are presented in Table 4. As can be seen, the weak supervision

[10] http://www.freebase.com/

(MLN) gets better performance than the distant supervision (Freebase) in both AUC ROC and AUC PR, which means that the supervision provided by the CMLN results in examples of higher quality. This observation is similar to the one made by Riedel et al. [19], as the Freebase is not necessarily the source of the NYT corpus (or vice-versa), the use of this knowledge base (distant supervision) makes a stronger assumption than our inferred "soft" examples. Using soft (probabilistic) labels is beneficial as opposed to a fixed label of a relation as the latter can possibly overfit.

| | AUC ROC | AUC PR |
|---|---|---|
| MLN | **0.53** | **0.56** |
| Freebase | 0.48 | 0.52 |

**Table 4.** Weak (MLN) & distant supervision (Freebase) results.

We also compared the best F1 value of our approach against the state-of-the-art distant supervision approach [22], which only learns the parameters for a hand-written structure. This is to say that the approach of Surdeanu et al. [22] assumes that the structure of the model is provided and hence simply learns the parameters of the model. On the other hand, our learning algorithm learns the model as well as its parameters. We obtained the code[11] for Surdeanu et al.'s approach and modified it to focus only on the *nationality* relation in both learning and inference. The best F1 values for this approach (MIML-RE) and our approach (MLN) are presented in Table 5. We report the best F1 values as this metric was reported in their earlier work.

| | Best F1 |
|---|---|
| MIML-RE | 0.16 |
| MLN | 0.14 |

**Table 5.** Best F1 value for MIML-RE and our approach for the *nationality* relation.

## 5 Conclusion

One of the key challenges for applying learning methods in many real-world problems is the paucity of good quality labeled examples. While semi-supervised learning methods have been developed, we explore another alternative method of weak supervision – where the goal is to create examples of a quality that can be relied upon. We considered the NLP tasks of relation extraction and document extraction to demonstrate the usefulness of the weak supervision. Our key insight is that weak supervision can be provided

[11] http://nlp.stanford.edu/software/mimlre.shtml

by a “domain” expert instead of a “NLP” expert and thus the knowledge is independent of the underlying problem but is close to the average human thought process – for example, sports fans. We are exploiting domain knowledge that the authors of articles assume their readers already know and hence the authors do not state it. We used the weighted logic representation of Markov Logic networks to model the expert knowledge, infer the relations in the unannotated articles, and adapted functional gradient boosting for predicting the target relations. Our results demonstrate that our method significantly improves the performance thus reducing the need for gold standard examples.

Our proposed method is closely related to distant supervision methods. So it will be an interesting future direction to combine the distant and weak supervision examples for structure learning. Combining weak supervision with advice taking methods [24, 2, 11] is another interesting direction. This method can be seen as giving advice about the examples, but AI has a long history of using advice on the model, the search space and examples. Hence, combining them might lead to a strong knowledge based system where the knowledge can be provided by a domain expert and not a AI/NLP expert. We envision that we should be able to ”infer” the world knowledge from knowledge bases such as Cyc or ConceptNet and employ them to generate the weak supervision examples. Finally, it is important to evaluate the proposed model in similar tasks.

## 6 Acknowledgements

Sriraam Natarajan, Tushar Khot, Jose Picado, Chris Re, and Jude Shavlik gratefully acknowledge support of the DARPA Machine Reading Program and DEFT Program under the Air Force Research Laboratory (AFRL) prime contract no. FA8750-09-C-0181 and FA8750-13-2-0039 respectively. Any opinions, findings, and conclusion or recommendations expressed in this material are those of the authors and do not necessarily reflect the view of the DARPA, AFRL, or the US government. Kristian Kersting was supported by the Fraunhofer ATTRACT fellowship STREAM and by the European Commission under contract number FP7-248258-First-MM.

## References

1. M. Craven and J. Kumlien. Constructing biological knowledge bases by extracting information from text sources. In *ISMB*, 1999.
2. S. Devlin, D. Kudenko, and M. Grzes. An empirical study of potential-based reward shaping and advice in complex, multi-agent systems. *Advances in Complex Systems*, 14(2):251–278, 2011.
3. T.G. Dietterich, A. Ashenfelter, and Y. Bulatov. Training conditional random fields via gradient tree boosting. In *ICML*, 2004.
4. P. Domingos and D. Lowd. *Markov Logic: An Interface Layer for AI.* Morgan & Claypool, San Rafael, CA, 2009.
5. R. Hoffmann, C. Zhang, X. Ling, L. Zettlemoyer, and D. S. Weld. Knowledge-based weak supervision for information extraction of overlapping relations. In *ACL*, 2011.
6. D. Jain. Knowledge engineering with Markov Logic Networks: A review. In *KR*, 2011.
7. R. Kate and R. Mooney. Probabilistic abduction using Markov Logic Networks. In *PAIR*, 2009.

8. K. Kersting and K. Driessens. Non–parametric policy gradients: A unified treatment of propositional and relational domains. In *ICML*, 2008.
9. T. Khot, S. Natarajan, K. Kersting, and J. Shavlik. Learning Markov logic networks via functional gradient boosting. In *ICDM*, 2011.
10. J. Kim, T. Ohta, S. Pyysalo, Y. Kano, and J. Tsujii. Overview of BioNLP'09 shared task on event extraction. In *BioNLP Workshop Companion Volume for Shared Task*, 2009.
11. G. Kuhlmann, P. Stone, R. J. Mooney, and J. W. Shavlik. Guiding a reinforcement learner with natural language advice: Initial results in robocup soccer. In *AAAI Workshop on Supervisory Control of Learning and Adaptive Systems*, 2004.
12. M. Mintz, S. Bills, R. Snow, and D. Jurafsky. Distant supervision for relation extraction without labeled data. In *ACL and AFNLP*, 2009.
13. S. Natarajan, T. Khot, K. Kersting, B. Guttmann, and J. Shavlik. Gradient-based boosting for statistical relational learning: The relational dependency network case. *Machine Learning*, 86(1):25–56, 2012.
14. J. Neville and D. Jensen. Relational dependency networks. In L. Getoor and B. Taskar, editors, *Introduction to Statistical Relational Learning*, pages 653–692. 2007.
15. F. Niu, C. Ré, A. Doan, and J. W. Shavlik. Tuffy: Scaling up statistical inference in Markov logic networks using an RDBMS. *PVLDB*, 4(6):373–384, 2011.
16. H. Poon and L. Vanderwende. Joint inference for knowledge extraction from biomedical literature. In *NAACL*, 2010.
17. S. Raghavan and R. Mooney. Online inference-rule learning from natural-language extractions. In *International Workshop on Statistical Relational AI*, 2013.
18. S. Riedel, H. Chun, T. Takagi, and J. Tsujii. A Markov logic approach to bio-molecular event extraction. In *BioNLP*, 2009.
19. S. Riedel, L. Yao, and A. McCallum. Modeling relations and their mentions without labeled text. In *ECML PKDD*, 2010.
20. S. Sorower, T. Dietterich, J. Doppa, W. Orr, P. Tadepalli, and X. Fern. Inverting grice's maxims to learn rules from natural language extractions. In *NIPS*, pages 1053–1061, 2011.
21. M. Surdeanu and M. Ciaramita. Robust information extraction with perceptrons. In *NIST ACE*, 2007.
22. M. Surdeanu, J. Tibshirani, R. Nallapati, and C. Manning. Multi-instance multi-label learning for relation extraction. In *EMNLP-CoNLL*, 2012.
23. S. Takamatsu, I. Sato, and H. Nakagawa. Reducing wrong labels in distant supervision for relation extraction. In *ACL*, 2012.
24. L. Torrey, J. Shavlik, T. Walker, and R. Maclin. Transfer learning via advice taking. In *Advances in Machine Learning I*, pages 147–170. 2010.
25. M. Verhagen, R. Gaizauskas, F. Schilder, M. Hepple, G. Katz, and J. Pustejovsky. SemEval-2007 task 15: TempEval temporal relation identification. In *SemEval*, 2007.
26. K. Yoshikawa, S. Riedel, M. Asahara, and Y. Matsumoto. Jointly identifying temporal relations with Markov logic. In *ACL and AFNLP*, 2009.
27. G. Zhou, J. Su, J. Zhang, and M. Zhang. Exploring various knowledge in relation extraction. In *ACL*, 2005.