# Minimally Supervised Domain-Adaptive Parse Re-ranking for Relation Extraction

**Feiyu Xu, Hong Li, Yi Zhang, Hans Uszkoreit, Sebastian Krause**

DFKI, LT-Lab, Germany
{feiyu,lihong,Yi.Zhang,uszkoreit,sebastian.krause}@dfki.de

## Abstract

The paper demonstrates how the generic parser of a minimally supervised information extraction framework can be adapted to a given task and domain for relation extraction (RE). For the experiments a generic deep-linguistic parser was employed that works with a largely hand-crafted head-driven phrase structure grammar (HPSG) for English. The output of this parser is a list of $n$ best parses selected and ranked by a MaxEnt parse-ranking component, which had been trained on a more or less generic HPSG treebank. It will be shown how the estimated confidence of RE rules learned from the $n$ best parses can be exploited for parse re-ranking. The acquired re-ranking model improves the performance of RE in both training and test phases with the new *first* parses. The obtained significant boost of recall does not come from an overall gain in parsing performance but from an application-driven selection of parses that are best suited for the RE task. Since the readings best suited for successful rule extraction and instance extraction are often not the readings favored by a regular parser evaluation, generic parsing accuracy actually decreases. The novel method for task-specific parse re-ranking does not require any annotated data beyond the semantic seed, which is needed anyway for the RE task.

## 1 Introduction

Domain adaptation is a central research topic for many language technologies including information extraction (IE) and parsing (e.g., (Grishman and Sundheim, 1996; Muslea, 1999; Hara et al., 2005; McClosky et al., 2010; Miwa et al., 2010)). The largest challenge is to develop methods that exploit domain knowledge with minimal human effort.

Many IE systems benefit from combining generic NLP components with task-specific extraction methods. Various machine learning approaches have been employed for adapting the IE methods to new domains and extraction tasks (e.g., (Yangarber, 2001; Sudo et al., 2003; Greenwood and Stevenson, 2006)). The IE framework extended in this paper utilizes minimally supervised learning of extraction rules for the detection of relation instances (Xu et al., 2007). Since the minimally supervised learning starts its bootstrapping from a few semantic examples, no treebanking or any other annotation is required for new domains. In addition to this inherently domain-adaptable rule-learning component, the framework also employs two language analysis modules: a named-entity (NE) recognizer (Drozdzynski et al., 2004) and a parser (Lin, 1998; de Marneffe and Manning, 2008). NE recognizers are adapted to new domains–if needed–by adding rules for new NE types and extending the gazetteers. The employed generic data-driven dependency parsers or deep-linguistic handcrafted parsers have not yet been adapted to IE domains and tasks.

The new work presented here concerns the adaptation of a generic parser to a given relation extraction (RE) task and domain without actually changing the parser itself. For the experiments a generic deep-linguistic parser was used together with a hand-crafted HPSG (Pollard and Sag, 1994) grammar for English (ERG) (Flickinger, 2000). The output of this parser is a list of $n$ best parses selected and ranked by a MaxEnt parse-ranking component (Toutanova et al., 2005b), which had been trained on a generic HPSG treebank (Oepen et al., 2002). The parse ranking had attracted our attention because the first RE tests with the handcrafted grammar revealed recall problems even for the parsable relation mentions. Our suspicion to partially blame the generic parse selection was confirmed by our experiments.

In this paper we will show how the estimated confidence of rules learned from the $n$ best parses can be exploited for task-specific parse re-ranking. The acquired re-ranking model improves the performance of RE both in training and test phases. The task-driven re-ranking leads to significantly better RE recall by boosting readings that are better suited for RE rule extraction and rule application. The beneficial re-ranking does not improve the quality of parsing measured by task-independent performance criteria, not even for the IE domain. The validation of the adapted parser using a hand-checked HPSG treebank of in-domain texts rather shows a deterioration of parsing accuracy. But often the incorrect parses selected over less faulty parses support the correct detection of instance mentions.

The novel method for task-specific parse re-ranking does not require any annotated data beyond the semantic seed, needed anyway for the RE task. Thus it does not require a domain-specific treebank.

The paper is organized as follows. Section 2 describes the grammar and the associated parse selection model. Section 3 introduces the RE framework. Section 4 explains the new task/domain-oriented re-ranking approach. Section 5 presents the experiments and evaluations. Special emphasis is placed on the role of re-ranking for the performance of the RE system. Section 6 discusses related work. Finally, Section 7 summarizes the results and suggests directions for further research.

## 2 HPSG and Parse Selection Model

Recent progress in parsing has several sources. The most noticeable trend is the shift from pure symbolic rule-based approaches toward statistical parsing. The availability of large-scale treebanks has enabled the training of powerful data-driven parsers, some based on constituency others on dependency. Meanwhile, existing hand-crafted precision oriented linguistic grammars have also benefitted from empirical methods through new disambiguation models trained on treebanks.

Among the available deep linguistic grammars, ERG is a good representative of the state of the art. Its lexicon contains ~35K entries. The 1004 release of the grammar we use is accompanied by a maximum-entropy-based parse disambiguation model trained on the Redwoods Treebank (Oepen et al., 2002), a treebank of ~20K sentences with mixed genre texts (dialogs, tourist information, emails, etc). The discriminative log-linear disambiguation model scores each parse by the following (Toutanova et al., 2005b),

$$P(t|w) = \frac{\exp \sum_{i=1}^{n} \lambda_i f_i(t, w)}{\sum_{t' \in T(w)} \exp \sum_{i=1}^{n} \lambda_i f_i(t', w)} \quad (1)$$

where $w$ is the given input sentence and $t$ is the HPSG reading; $T(w)$ is the set of all possible readings for a given sentence $w$ licensed by the grammar; $\langle f_1, \ldots, f_n \rangle$ and $\langle \lambda_1, \ldots, \lambda_n \rangle$ are feature functions and their corresponding weights. In practice, the effective features are defined on the HPSG derivation trees (without details from the feature structures), and the best readings are decoded efficiently from a packed parse forest with dynamic programming (Zhang et al., 2007).

Although there are indications that parsers with hand-written grammars usually suffer less from the shift of domain than statistical parsers (Zhang and Wang, 2009; Plank and van Noord, 2010), the effect can still be observed, say in the preference of lexical selection. The issue is not that the correct analysis would be ruled out by the constraints in the treebank-induced grammar, but rather that it is not favored by the statistical ranking model, since the statistical distribution of the syntactic structures in the training corpus is different from the target application domain. This issue is recently acknowledged in most parsing systems and known as the domain adaptation task.

## 3 DARE and Confidence Estimation

DARE (Xu et al., 2007; Xu, 2007) is a minimally supervised machine learning system for RE for free texts consisting of two major parts: 1) rule learning, 2) relation extraction (RE). DARE provides a recursive extraction-rule representation, which can deal with relations of varying complexity. Rule learning and RE feed each other in a bootstrapping framework. The bootstrapping starts from so-called ”semantic seed” as a search query, which is a small set of instances of the target relation. The rules are extracted from found sentences with annotations of semantic entities and parsing results. RE applies acquired rules to texts in order to discover more relation instances, which in turn are employed as seed for further iterations. The confidence values of the newly acquired rules and instances are calculated in the spirit of the "Duality principle" (Brin, 1998; Agichtein and Gravano, 2000; Yangarber, 2001), i.e., the confidence values of the rules are dependent on the truth value of their extracted instances and on the seed instances from which they stem. The confidence value of an extracted instance makes use of the confidence value of its ancestor seed instances. The core system architecture of DARE is depicted in Figure 1. The entire bootstrapping stops when no new rules or new instances can be detected.

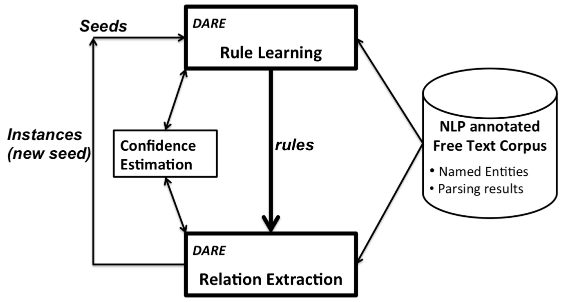

Figure 1: DARE core architecture

Relying entirely on semantic seeds as domain knowledge, DARE can accommodate new relation types and domains with minimal effort. Since we had already reported on experiments applying the framework to different relation types and corpora including MUC-6 data in the cited papers, including comparisons with other ML approaches to RE (Xu, 2007; Uszkoreit et al., 2009), we omit a comparative discussion here.

For confidence estimation, the method proposed by Xu et al. (2010) is adopted.[1] Actually, in (2) we propose an extended version of the rule scoring, since the rule scoring in (Xu et al., 2010) did not consider the case when a learned rule does not extract any new instances. Thus, given the scoring of instances, the confidence value of a rule is the average score of all instances ($\mathbb{I}_{extracted}$) extracted by this rule or the average score of seed instances ($I_{rule}$) from which they are learned. Through the factor $\delta$ we reduce the score of rules that have not proven yet their potential for extracting instances.

$$\begin{aligned}
&\textbf{confidence}(rule) = \\
&\quad\begin{cases}
\dfrac{\sum_{i \in \mathbb{I}_{extracted}} \textbf{score}(i)}{|\mathbb{I}_{extracted}|} & \text{if } \mathbb{I}_{extracted} \neq \phi \\[2ex]
\dfrac{\sum_{j \in I_{rule}} \textbf{score}(j)}{|I_{rule}|} \times \delta & \text{if } \mathbb{I}_{extracted} = \phi
\end{cases} \\
&\text{where} \quad \mathbb{I}_{extracted} = \textbf{getInstances}(rule), \\
&\qquad\qquad I_{rule} = \textbf{getMotherInstancesOf}(rule), \\
&\qquad\qquad \delta = 0.5
\end{aligned} \tag{2}$$

This method allows DARE to estimate the confidence value of a rule according to its extraction performance or the confidence value of its origin.

# 4 Domain Adaptive Parse Re-Ranking

## 4.1 Basic Idea

In our research, we observe that there is a strong connection between the RE task and the parser via the learned extraction rules, because these rules are derived from the parse readings. The confidence values of the extraction rules imply the domain appropriateness of the parse readings. Therefore, the confidence values can be utilized as feedback to the parser to help it to re-rank its readings.

## 4.2 Re-Ranking Architecture and Method

Figure 2 depicts the overall architecture of our experimental system. We utilize the HPSG to parse our experimental corpus and keep the first $n$ readings of each sentence (e.g., 256) delivered by the parser. During bootstrapping DARE tries to learn extraction rules from all readings of sentences containing a seed instance or newly detected instances. At each iteration the extracted rules are applied to all readings of all sentences. When bootstrapping has terminated, the obtained rules are assigned confidence values based on the DARE ranking method described in Section 3.

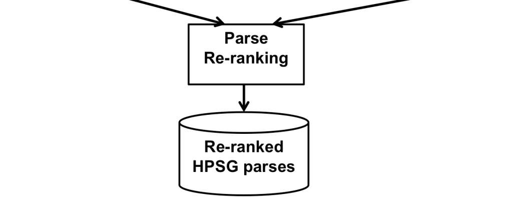

Figure 2: DARE and Parse Re-ranking

[1] The actual confidence estimation is slightly more complex because it further improves the scoring by utilizing implicit negative evidence provided by closed-world seeds, a method proposed by (Xu et al., 2010). As this mechanism is not relevant in the context of this paper, we omit a description.

The parse re-ranking component scores the alternative parses of each sentence based on the confidence values of the rules matching these parses, i.e., all rules that could have been extracted from a parse or successfully applied to it.

For each reading from the HPSG parser, the re-ranking model assigns a numeric score by the following formula:

$$S(t) = \begin{cases} \sum_{r \in R(t)} (\mathbf{confidence}(r) - \phi\mathbf{confidence}) & if R(t) \neq \phi, \\ 0 & if R(t) = \phi. \end{cases} \tag{3}$$

$R(t)$ is the set of RE rules matching parse reading $t$, and $\phi\mathbf{confidence}$ is the average confidence score among all rules. The score of the reading will be increased if the matching rule has an above-average confidence score. And the matching of low-confidence rules will decrease the reading's re-ranking score. If a reading has no matching DARE rule, it will be assigned the lowest score 0, for no potential relation can be extracted from that reading.

After the calculation, the top-$n$ readings are sorted in descending order. In case two or more readings received the same re-ranking score (e.g. by matching the same set of DARE rules), the original maximum entropy-based disambiguation scores are used as a tie-breaker. The sort comparison function is shown below:

**Algorithm 1** compare_readings($r_i$, $r_j$)

```
if compare($S(r_i)$, $S(r_j)$) $\neq 0$ then
    return compare($S(r_i)$, $S(r_j)$)
else # Tie-breaking with MaxEnt scores
    return compare($MaxEnt(r_i)$, $MaxEnt(r_j)$)
end if
```

In practice, most readings will have no more than two matching DARE rules. And many readings from the HPSG parser do not affect the RE task. A consequence is that the re-ranking model can only partially disambiguate and have an effect only on particular subsets of the readings. As we are only evaluating RE performance, the remaining ambiguity is not an issue.

# 5 Experiments and Evaluation

## 5.1 Experiment and Evaluation Setup

**Data** For several reasons we decided to conduct our experiments on the Nobel Prize award corpus used also in (Xu et al., 2007). Previous results have shown that

1. not every data collection is suited for the minimally supervised approach to RE (Xu, 2007);

2. freely available Nobel Prize award corpus actually has the required properties (Uszkoreit et al., 2009).

Moreover, the availability of a version of the corpus in which all relation mentions are labelled and a treebank for a subset of the corpus have greatly facilitated the evaluation.

The target relation is *prize-awarding event*, namely, a relation among four arguments: WINNER, PRIZE_NAME, PRIZE_AREA and YEAR. We take the same seed example as utilized in (Xu et al., 2007), namely, the 1999 Nobel Chemistry winner Ahmed H Zewail in our experiments[2]. The seed looks like an database recond:

$\langle Ahmed\ H\ Zewail, Nobel, Chemistry, 1999\rangle$

The corpus contains 2864 documents from BBC, CNN and NYT, together 143289 sentences. ERG covers around 70% sentences of the total corpus. For our experiments we randomly divide the parsable corpus into two parts: training and test corpus, each containing the same number of sentences. The average sentence length of the total corpus is around 20 words. If we look at the domain relevant sentences, namely, those contain both person name mentions and prize name mentions, they have an average length of around 30. Among those relevant ones, the average length of the sentences parsable by ERG is around 25.

**Experiments** Two phases of experiments are conducted. In the *training* phase, we show that re-ranking improves RE performance. The *test* phase applies the re-ranking model resulting from the training phase to the test corpus. In both phases, two different experiments are conducted *1) Baseline*: without re-ranking; *2) Re-ranking*: with parse re-ranking. In the baseline experiment,

[2] Uszkoreit et al. (2009) show that for the given dataset the particular choice of the single seed instance does not have any affect on the performance.

we keep the first $n$ readings of all sentences and run DARE for rule learning and RE on top of these readings. The aim is to observe whether correct relation instances can also be detected in lower-ranked readings. In the second experiments, we aim to investigate whether re-ranking based on task-feedback and domain knowledge is useful for better extraction performance. These experiments are conducted only with the best reading after re-ranking, i. e. the normal setting of RE application. In none of the experiments, confidence thresholds are employed for improving precision by filtering out less confident rules or instances. As we are mainly interested in the effects of re-ranking on RE recall, we are trying to avoid any other factors that may influence the recall. Thus in our experiments confidence estimation scores are only used for re-ranking.

**Qualitative Analysis** Given the experimental results, we carry out various qualitative analysis on the results of both parsing and RE. With respect to parsing, we evaluate the results against the gold-standard treebank before and after re-ranking. In addition we evaluate the quality of the extraction rules before and after the re-ranking.

## 5.2 Experiments

### 5.2.1 Training

**Baseline** Figure 4 shows the baseline evaluation results. In this case, no confidence thresholds are applied, therefore we have neither re-ranking nor filtering. In order to monitor the contribution of lower-ranked parses to RE, we add readings in logarithmic increments. We start with one reading, namely the best reading proposed by the parser and then in steps go up to 500. From each reading, DARE tries to learn rules and to extract relation instances. When DARE only works with the best reading, the precision is very high, namely, 87.83%, but with a very low recall of 45.18%. When we increase the number of readings, we observe that precision drops while recall increases. This confirms our suspicion that many good readings are among the lower ranked ones in the current maximum entropy-based parse model. Therefore, re-ranking is important for lifting the good readings to the top.

**Re-ranking** In the training phase, we learn DARE rules from all 500 readings from all sentences in the training corpus. Given the rules and their confidence values, we re-rank the 500 readings of each sentence in this corpus.

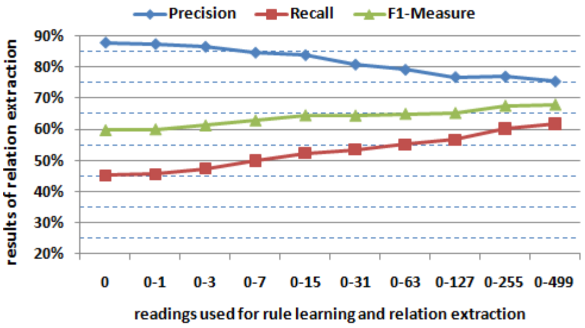

Figure 4: Training phase (baseline): RE performance w.r.t. the increase of readings

| Reading 0 | Precision | Recall | F1-Measure |
|---|---|---|---|
| Baseline (no re-ranking) | 87.83% | 45.18% | 59.66% |
| After re-ranking | 83.87% | 56.19% | 67.29% |

Table 1: Training phase: Comparison of RE performance before and after re-ranking.

Table 1 compares the RE performance with just the first reading before re-ranking (baseline experiment) and after re-ranking. As indicated, the re-ranking strongly improves the recall value (56.19% vs. 45.18%) and also yields a significantly better F-measure (67.29% vs. 59.66%).

Figure 5 illustrates the behavior of parse readings with respect to the respective frequencies of matches with extraction rules (indicating their usefulness for rule or instance extraction). After re-ranking, the number of the higher ranked readings that match with the RE rules is increased significantly. This indicates that the higher ranked readings after re-ranking are better suited for the RE task.

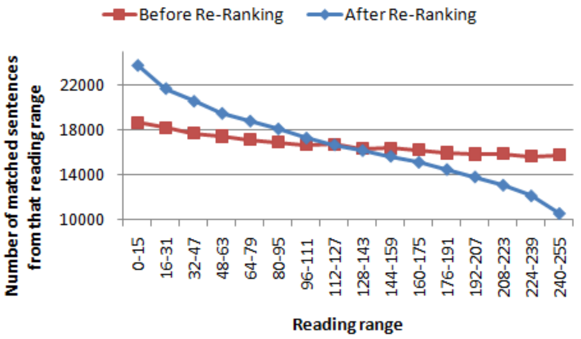

Figure 5: Training phase: Distribution of parse readings from 0 to 255 and their frequency of matching rules before and after re-ranking

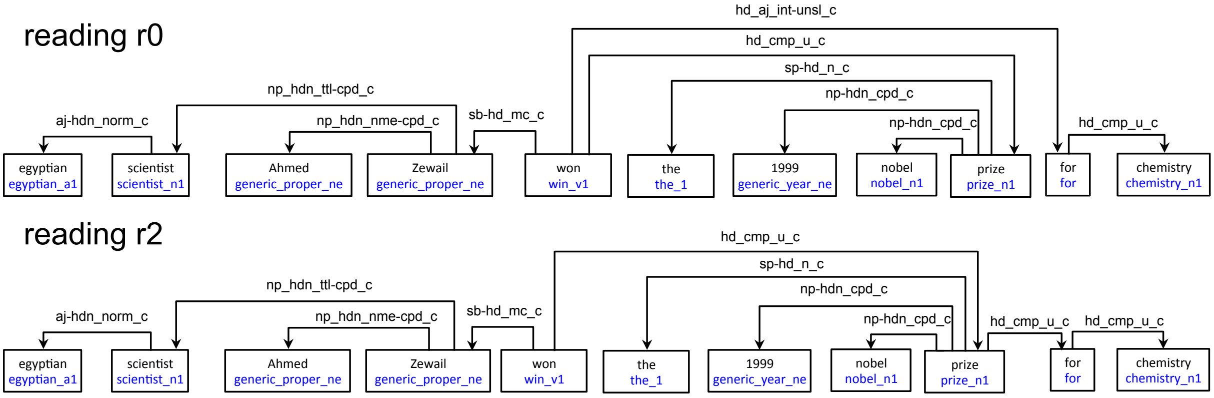

Figure 3: An example of ambiguous parses with PP attachment

**Re-ranking Examples** In our experiment, we utilize the syntactic derivation tree of the HPSG analysis. Figure 3 shows two derivation trees of a sentence (4) from the experimental domain corpus.

(4) *Egyptian scientist Ahmed Zewail won the 1999 Nobel Prize for Chemistry*

In Figure 3, in the first reading *r0* the PP *"for chemistry"* is wrongly attached to the verb *"win"*, while *r2* (the third reading) is more appropriate since the PP here modifies the noun *"prize"*. The DARE rule in Figure 6 is presented as a typed feature structure, which is learned from HPSG parses. The value of its feature *PATTERN* contains the derivation tree structures relevant for the target relation, while the value of the feature *OUTPUT* represents the co-indexing between the semantic arguments of the target relation and the linguistic arguments in *PATTERN*. Since this rule has a high confidence value and it matches the reading *r2*, *r2* is pushed to the top after re-ranking.

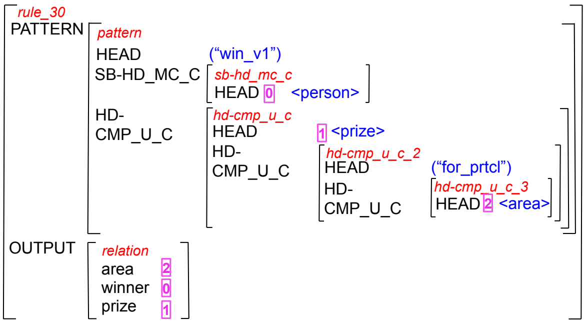

Figure 6: An example DARE rule derived from HPSG derivation trees

### 5.2.2 Testing

In the test phase, we apply the re-ranking model trained in the training phase to the parsing of the test corpus when performing RE. The re-ranking model consists of RE rules with their respective confidence values. These rules work as classifiers that add their confidence values to the ranking scores of matching readings.

**Baseline** First, we evaluate the performance of the baseline system, i.e., parsing the test corpus without re-ranking. Similar to the experiments on the training corpus, we first examine the performance of RE on different reading sets. The results are shown in Figure 7. Similar to the training phase results, the recall and F-measure values increase when more readings are taken into account.

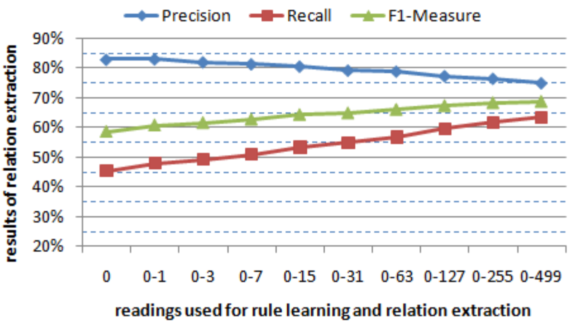

Figure 7: Test phase (baseline): RE performance with respect to the increase of readings.

**Re-ranking** Table 2 presents the extraction performance after application of the trained re-ranking model to the test corpus, using only the highest-ranked reading. Similar to the training phase results, both recall and F-measure also improve significantly in comparison to the baseline system before re-ranking.

| Reading 0 | **Precision** | **Recall** | **F1-Measure** |
|---|---|---|---|
| Baseline (no reranking) | 82.93% | 45.37% | 58.56% |
| after re-ranking | 80.33% | 53.41% | 64.16% |

Table 2: Test phase: Comparison of RE performance before and after re-ranking.

## 5.3 Qualitative Analysis

Experiments in both training and test phases confirm that our re-ranking improves RE recall and F-measure. A further observation is that the re-ranked best readings are much more compatible with the learned extraction rules. Naturally, the question arises whether re-ranking also improves overall parsing accuracy.

### 5.3.1 Parsing before and after Re-ranking

Finally, we evaluate the general parsing accuracy before and after re-ranking. More specifically, we compare the syntactic structures against a high-quality gold-standard treebank annotated by the ERG grammar developer. This evaluation indicates the general correctness of the parser (or in particular the disambiguation model).[3]

Table 3 reveals that the general parsing performance suffers from re-ranking both with respect to full trees and subtrees. To further narrow down the effect of re-ranking, we manually marked the regions (sub-strings in sentences) most relevant for the target relations and calculated the parser scores within those subtrees.[4] The degradation of parser performance (against gold annotation) is more significant within these local regions.

| Model | $LB_{f_1}(full)$ | $LB_{f_1}(subtree)$ |
|---|---|---|
| MaxEnt | 0.8613 | 0.8918 |
| Reranked | 0.7966 | 0.8132 |

Table 3: Labeled bracketing f-score

Further error analyses show the breakdown of the differences: Of the 113 test sentences, 68 show a difference w.r.t. re-ranking. The labeled bracketing accuracy (on relevant subtrees) increased for 13 sentences. Among these, 3 are due to better *appositions*, 2 to better selection of *verb subcat frames*, 6 to better *PP attachments*. Of the 55 cases of degradation, main causes are: incorrect *compounding in NPs* (24 cases), bad *coordinations* (7 cases), wrong *lexical categories* (2 cases).

| | “good” for RE |
|---|---|
| Before re-ranking | 50% |
| After re-ranking | 85% |

Table 4: “Good” readings for RE among 68 reranked sentences

A careful study has been conducted on these 68 cases with respect to their effect on RE performance. It turns out that after re-ranking more of the parses are “good” for RE, i.e., leading to good rules. A “good” rule is defined by us as a rule which extracts correct instances. Table 4 shows that after re-ranking 85% of the 68 have good parses as opposed to 50% before re-ranking.

An explanation for the drop of linguistic quality is that linguistically “wrong” analyses nevertheless lead to consistent extraction of rules and instances. For example, the gold-standard bracketing of the compound noun “Nobel Peace Prize laureate” is *((Nobel (Peace Prize)) laureate)*. The re-ranking reading is *((Nobel Peace) (Prize laureate))*, which is wrong. However, the rule derived from this wrong reading can be applied to all equally incorrect readings of similar compound nouns such as “Nobel Chemistry/Physics/Economics Prize laureate” to successfully extract two arguments of the target relation, namely, PRIZE_NAME and PRIZE_AREA. Thus the increased consistency in the re-ranked parses does help improve the RE process.

### 5.3.2 Extraction Rules after Re-ranking

In the above analysis, we can learn the lessons that a good reading for RE task is not necessary a linguistically correct parse. The major contribution of re-ranking is not the improvement of general linguistic parse selection but the improvement of selection of good readings for RE tasks.

Table 5 shows a comparison of the distribution of the good readings before and after re-ranking in

[3] Since manual treebanking of HPSG derivation trees is very expensive, the gold-standard treebank only contains 500 randomly selected domain relevant sentences in which both persons and prizes are mentioned. Among these 500 sentences, 113 are in the test corpus. Although this treebank was developed independent from our research approach, the 113 sentences turn out to be useful because they are potential candidates for RE rules and thus their readings can be more effected by re-ranking than sentences which are irrelevant for the target relation.

[4] We also evaluated the parsing performance on the subtrees selected by the relation extraction rules, whose results are consistent with the above findings.

test corpus. Bad readings are readings where bad rules are learned, namely, rules which extract only incorrect instances. Useless readings are readings from which useless rules are learned. Useless rules are rules which do not extract any instance. Table 5 clearly demonstrates that the porportion of good readings increases significantly after re-ranking, while the number of bad readings and useless readings drop.

| | Good Reading | Bad Reading | Useless Reading |
|---|---|---|---|
| before re-ranking | 29.2% | 1.3% | 69.5% |
| after re-ranking | 42.4% | 0.8% | 56.8% |

Table 5: Test corpus: distribution of good readings before and after re-ranking

We also compare the number of the learned good rules and their extraction productivity. After re-ranking, not only the number of good rules increases, but also the average number of the instances extracted by each good rule is grown to 4.3 in comparison to 3.5 before re-ranking. The growth of good readings and rules and the productivity of rule extraction performance explains the recall improvement after the parse re-ranking.

## 6 Related Work

Various attempts have been made to improve the cross-domain performance of statistical parsing models. McClosky et al. (2006) uses self-training to improve Charniak's parser by feeding large amount of unannotated texts to the parser. Plank (2009) utilize structural-correspondence learning to improve the accuracy of the Dutch Alpino parser on the Wikipedia texts. Rimell and Clark (2008) show that a small set of annotated in-domain data can significantly improve the CCG parser's performance. Hara et al. (2007) improves the Enju HPSG parser performance in the biomedical domain by a low-cost retraining of the lexical disambiguation model. Nearly all approaches evaluate the parsing quality against a "gold-standard" treebank. Miwa et al. (2010) compares five parsers for bio-molecular event extraction to investigate the correlation between the performance on a gold-stand treebank and the usefulness in real-world applications. All four domain-adapted parsers achieve similar IE performance and are better than the one not adapted.

The idea of re-ranking parses for better disambiguation is not new. Charniak and Johnson (2005) presents a discriminative model for capturing the linguistically motivated global properties of the candidate parses proposed by the first-stage generative parser. As the re-ranking model operates on a relatively small set of candidates, it is able to more accurately find the best reading. In the same spirit, several applications such as named-entity extraction (Collins, 2002), semantic parsing (Toutanova et al., 2005a) and semantic labeling (Ge and Mooney, 2006) have taken advantage of re-ranking approaches based on discriminative models.

In contrast to the above proposals, our approach does not need the annotated "gold-standard" data for domain adaptation or training of the re-ranking model. Our system exploits application feedback for re-ranking. In a sense, the approach is akin in spirit to the joint learning of multiple types of linguistic structures with non-jointly labeled data (Finkel and Manning, 2010), although in our case the emphasis is entirely put on the application performance.

## 7 Conclusion and Future Work

The main contribution of our work is a method for adapting generic parsers to the tasks and domains of relation extraction by parse re-ranking. Our re-ranking is based on feedback from the application. We could show that for one generic parser/grammar, recall and f-measure could be considerably improved and hope that this effect can also be obtained for other generic parsers. We do not worry much about the collateral decrease in precision, because precision will be tightened again when we employ confidence estimation thresholds for filtering out less promising rules and instances.

A side result of the work was the insight that a better parse ranking for the purpose of relation extraction does not necessarily correspond to a better parse ranking for other purposes or for generic parsing. This should not be surprising since relation extraction in contrast to text understanding does not need the entire and correct syntactic structure for the detection of relation instances. The ease and consistency of rule extraction and rule application counts more than the linguistically correct analysis. The gained new insight that the consistency of parse selection is more relevant than parsing accuracy, we consider worth sharing.

The presented results may also be viewed as a step forward toward making deep linguistic grammars useful for relation extraction, whereas up to now most minimally supervised approaches to RE have employed shallower robust parsers. The hope behind these attempts is to improve precision without losing too much recall. After reclaiming recall through our parse re-ranking, next steps in this line of research will be dedicated to balancing off the deficits in coverage by data-driven lexicon extension in the spirit of (Zhang et al., 2010) and by exploiting the chart for partial parses involving the relevant types of named entities. Furthermore, the approach of (Dridan and Baldwin, 2010) to learning a parse selection model in an unsupervised way by utilizing the constraints of HSPG grammars might also be interesting for domain adaptive parse selection for relation extraction. At some point we may then be in a position to conduct a fair empirical comparison between deep-linguistic parsing with hand-crafted grammars on the one hand and purely statistical parsing on the other. An error analysis may then indicate the chances for hybrid approaches. However, before targeting these medium-term goals we plan to investigate whether our approach can also be applied to other parsers with inherent generic parse ranking and whether the set of learned RE rules with their confidence values can be directly used as features in the statistical parse disambiguation models instead of in the post-processing step by a separate re-ranker.

## Acknowledgements

This research was conducted in the context of the German DFG Cluster of Excellence on Multimodal Computing and Interaction (M2CI), projects Theseus Alexandria and Alexandria for Media (funded by the German Federal Ministry of Economy and Technology, contract 01MQ07016), and project TAKE (funded by the German Federal Ministry of Education and Research, contract 01IW08003).

## References

Eugene Agichtein and Luis Gravano. 2000. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the 5th ACM International Conference DL'00*, San Antonio, TX, June.

Sergey Brin. 1998. Extracting patterns and relations from the world wide web. In *WebDB Workshop at EDBT 98*.

Eugene Charniak and Mark Johnson. 2005. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *Proceedings of ACL 05*, pages 173–180, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Michael Collins. 2002. Ranking algorithms for named-entity extraction: Boosting and the voted perceptron. In *Proceedings of ACL '02*.

Marie-Catherine de Marneffe and Christopher D. Manning. 2008. The stanford typed dependencies representation. In *Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation at COLING08*, Manchester, UK.

Rebecca Dridan and Timothy Baldwin. 2010. Unsupervised parse selection for hpsg. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 694–704, Stroudsburg, PA, USA. Association for Computational Linguistics.

Witold Drozdzynski, Hans-Ulrich Krieger, Jakub Piskorski, Ulrich Schäfer, and Feiyu Xu. 2004. Shallow processing with unification and typed feature structures — foundations and applications. *Künstliche Intelligenz*, 1.

Jenny Rose Finkel and Christopher D. Manning. 2010. Hierarchical joint learning: Improving joint parsing and named entity recognition with non-jointly labeled data. In *Proceedings of ACL '10*, pages 720–728, Uppsala, Sweden.

Dan Flickinger. 2000. On building a more efficient grammar by exploiting types. *Natural Language Engineering*, 6(1):15–28.

Ruifang Ge and Raymond J Mooney. 2006. Discriminative reranking for semantic parsing. In

*Proceedings of COLING and ACL 06)*, pages 263–270. Association for Computational Linguistics.

Mark A. Greenwood and Mark Stevenson. 2006. Improving semi-supervised acquisition of relation extraction patterns. In *Proceedings of the Workshop on Information Extraction Beyond The Document*, pages 29–35, Sydney, Australia, July. Association for Computational Linguistics.

Ralph Grishman and Beth Sundheim. 1996. Message understanding conference - 6: A brief history. In *Proceedings of COLING 96*, Copenhagen, June.

Tadayoshi Hara, Yusuke Miyao, and Jun'ichi Tsujii. 2005. Adapting a probabilistic disambiguation model of an HPSG parser to a new domain. In *Proceedings of IJCNLP 05*.

Tadayoshi Hara, Yusuke Miyao, and Jun'ichi Tsujii. 2007. Evaluating impact of re-training a lexical disambiguation model on domain adaptation of an hpsg parser. In *Proceedings of the Tenth International Conference on Parsing Technologies*, pages 11–22, Prague, Czech Republic, June. Association for Computational Linguistics.

Dekan Lin. 1998. Dependency-based evaluation of MINIPAR. *Workshop on the Evaluation of Parsing Systems.*

David McClosky, Eugene Charniak, and Mark Johnson. 2006. Reranking and self-training for parser adaptation. In *Proceedings of COLING and ACL 06*, pages 337–344, Sydney, Australia.

David McClosky, Eugene Charniak, and Mark Johnson. 2010. Automatic domain adaptation for parsing. In *Proceedings of HLT and NAACL '10*, pages 28–36, Los Angeles, California, June. Association for Computational Linguistics.

Makoto Miwa, Sampo Pyysalo, Tadayoshi Hara, and Jun'ichi Tsujii. 2010. A comparative study of syntactic parsers for event extraction. In *Proceedings of the 2010 Workshop on Biomedical Natural Language Processing*, pages 37–45, Uppsala, Sweden, July. Association for Computational Linguistics.

Ion Muslea. 1999. Extraction patterns for information extraction tasks: A survey. In *AAAI Workshop on Machine Learning for Information Extraction*, Orlando, Florida, July.

Stephan Oepen, Kristina Toutanova, Stuart Shieber, Christopher Manning, Dan Flickinger, and Thorsten Brants. 2002. The LinGO Redwoods treebank: motivation and preliminary applications. In *Proceedings of COLING 2002: The 17th International Conference on Computational Linguistics: Project Notes*, Taipei, Taiwan.

Barbara Plank and Gertjan van Noord. 2010. Grammar-driven versus data-driven: Which parsing system is more affected by domain shifts? In *Proceedings of the 2010 Workshop on NLP and Linguistics: Finding the Common Ground*, Uppsala, Sweden, July.

Barbara Plank. 2009. A comparison of structural correspondence learning and self-training for discriminative parse selection. In *Proceedings of the NAACL HLT 2009 Workshop on Semi-supervised Learning for Natural Language Processing*, pages 37–42, Boulder, Colorado, June. Association for Computational Linguistics.

Carl J. Pollard and Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. University of Chicago Press, Chicago, USA.

Laura Rimell and Stephen Clark. 2008. Adapting a lexicalized-grammar parser to contrasting domains. In *Proceedings of EMNLP 08*, pages 475–484, Honolulu, Hawaii, October. Association for Computational Linguistics.

K. Sudo, S. Sekine, and R. Grishman. 2003. An improved extraction pattern representation model for automatic IE pattern acquisition. *Proceedings of ACL 2003.*

Kristina Toutanova, Aria Haghighi, and Christopher D. Manning. 2005a. Joint learning improves semantic role labeling. In *Proceedings of ACL 05*, page 589 596. Association for Computational Linguistics.

Kristina Toutanova, Christoper D. Manning, Dan Flickinger, and Stephan Oepen. 2005b. Stochastic HPSG parse selection using the Redwoods corpus. *Journal of Research on Language and Computation*, 3(1):83–105.

Hans Uszkoreit, Feiyu Xu, and Hong Li. 2009. Analysis and improvement of minimally supervised machine learning for relation extraction. In *14th International Conference on Applications of Natural Language to Information Systems*. Springer.

Feiyu Xu, Hans Uszkoreit, and Hong Li. 2007. A seed-driven bottom-up machine learning framework for extracting relations of various complexity. In *Proceedings of ACL 2007*, Prague, Czech Republic, 6.

Feiyu Xu, Hans Uszkoreit, Sebastian Krause, and Hong Li. 2010. Boosting relation extraction with limited closed-world knowledge. In *Proceedings of COLING '10, Poster Session*. Association for Computational Linguistics.

Feiyu Xu. 2007. *Bootstrapping Relation Extraction from Semantic Seeds*. Phd-thesis, Saarland University.

Roman Yangarber. 2001. *Scenarion Customization for Information Extraction*. Dissertation, Department of Computer Science, New York University, New York, USA.

Yi Zhang and Rui Wang. 2009. Cross-domain dependency parsing using a deep linguistic grammar. In *Proceedings of the Joint Conference ACL and AFNLP 09*, Suntec, Singapore, August.

Yi Zhang, Stephan Oepen, and John Carroll. 2007. Efficiency in unification-based N-best parsing. In *Proceedings of the 10th International Conference on Parsing Technologies (IWPT 2007)*, pages 48–59, Prague, Czech.

Yi Zhang, Timothy Baldwin, Valia Kordoni, David Martinez, and Jeremy Nicholson. 2010. Chart mining-based lexical acquisition with precision grammars. In *Proceedings of HLT and NAACL '10*, HLT '10, pages 10–18, Stroudsburg, PA, USA. Association for Computational Linguistics.