An Efficient Active Learning Framework for New Relation Types

Lisheng Fu
Computer Science Department
New York University
lf1099@nyu.edu

Ralph Grishman
Computer Science Department
New York University
grishman@cs.nyu.edu

Abstract

Supervised training of models for semantic relation extraction has yielded good performance, but at substantial cost for the annotation of large training corpora. Active learning strategies can greatly reduce this annotation cost. We present an efficient active learning framework that starts from a better balance between positive and negative samples, and boosts training efficiency by interleaving self-training and co-testing. We also studied the reduction of annotation cost by enforcing argument type constraints. Experiments show a substantial speed-up by comparison to the previous state-of-the-art pure co-testing active learning framework. We obtain reasonable performance with only 150 labels for individual ACE 2004 relation types.

1 Introduction

Relation extraction aims to discover the semantic relationship, if any, between a pair of entities in text. This structured information can be used to build higher-level applications such as question answering and other text mining applications.

Relation extraction was intensively studied as part of the multi-site ACE [Automatic Content Extraction] evaluations conducted in 2003, 2004, and 2005. For 2004, six major relation types were defined. Each relation mention takes two entity mention arguments in the same sentence. In annotating text, each entity mention pair within one sentence will be labeled if it involves one of the relation types. As part of ACE, substantial hand-annotated corpora marked with entities and relations were produced. For example, the ACE 2004 corpus had in total about 5,000 relation instances (and about 45,000 same-sentence entity pairs not bearing one of these relations). These large training corpora stimulated research on the supervised training of relation extractors, with considerable success: the best systems, when given hand-tagged entities, correctly identify and classify relations with an F score above 70% (Jiang and Zhai 2007).

Although supervised methods were effective, annotating a corpus of this size is too expensive in practice to serve as a model for developing new extractors: it requires consideration of 50K instances, of which only a small portion involve the target relation type. In consequence, most research has focused on reducing the annotation cost through semi-supervised learning methods such as bootstrapping systems. However, with limited labeled data, those semi-supervised systems failed to come close to the supervised level of performance. Their performance also varies with the distribution of seeds.

Recent studies have proposed new ways of reducing the annotation cost by using active learning. The advantage of active learning is that it can achieve reasonable performance, and even performance comparable to the supervised version, with few labeled examples, due to its ability to selectively sample unlabeled data for annotation.

To further reduce the annotation cost and provide an efficient framework for rapidly developing relation extraction models, we combine active learning with semi-supervised methods, provide solutions to the imbalanced seed set and uneven co-testing classifiers, and optionally incorporate argument type constraints. Most relation types now achieve reasonable performance with only 150 labeled instances. Section 2 gives more related work in detail. Section 3 describes the enhancements we have made. Section 4 reports the experimental results and the improvement in performance when only a few instances have been labeled. Section 5 concludes the paper.

2 Related Work

For reducing the cost of annotation in the task of relation extraction, most prior work used semi-

International Joint Conference on Natural Language Processing, pages 692–698,
Nagoya, Japan, 14-18 October 2013.

supervised learning. (Uszkoreit 2011) introduced a bootstrapping system for relation extraction rules, which achieved good performance under some circumstances. However, most previous semi-supervised methods have large performance gaps from supervised systems, and their performance depends on the choice of seeds (Vyas et al., 2009; Kozareva and Hovy, 2010).

Recent studies have shown the effectiveness of active learning for this task. (Zhang et al., 2012) proposed a unified framework for biomedical relation extraction. They used an SVM as the local classifier and tried both uncertainty-based and density-based query functions and showed comparable results for the two methods. They also proposed using cosine-distance to ensure the diversity of queries.

(Roth and Small 2008) used a dual strategy active learner (Donmez, Carbonell, & Bennett 2007) in their pipeline models of segmentation, entity classification and relation classification at the same time. They also adopted a regularized version of the structured perceptron (Collins 2002) instead of SVM and reported better results in active learning. Their work simulated the whole pipeline in active learning to achieve relation extraction, but had no specific research on the stage of relation extraction in the pipeline.

(Zhang 2010) proposed multi-task active learning with output constraints as a generalization of multi-view learning. The multi-task method relied on constraints on output between different tasks; this might be extended to situations where we need to learn relation sub-types as well as types, but was not applicable when relation extraction is an individual task.

Multi-view learning in a co-testing framework was used in (Sun and Grishman 2012). This paper proposed an LGCo-testing framework in which the local view is a maximum-entropy model with local features, and the global view is based on the distributional similarity in a large unlabeled corpus of the phrases between the two entity mentions of a relation. Extractor training was faster than with alternative active learning methods – much faster than with sequential annotation.

There has been research on combining different learning methods with active learning to obtain further improvement. (Song et al. 2011) used variants of SVM to apply semi-supervised learning after active learning in protein-protein interaction extraction.

The current paper adopts the earlier co-testing framework (Sun and Grishman 2012) and examines some of the design issues in order to achieve substantial further speed-ups.

3 Method

3.1 Framework

In active learning, users are asked to judge whether a particular sentence expresses the target relation between two entity mentions. For a fixed number of queries (fixed annotation cost), active learning aims to achieve the highest performance possible. The work described here builds on a state-of-the-art co-testing based active learning algorithm (Sun and Grishman 2012). Our framework starts with a better initial setting (section 3.2), and then interleaves self-training with querying (section 3.3). We adjust for imbalanced classifiers (section 3.4) to improve query selection. By enforcing entity type constraints (section 3.5), the annotation cost could be further reduced. This framework is able to build a bridge between labeled data and unlabeled data more rapidly than previous pure co-testing based active learning.

The overall procedure is as follows:

```
Let:
U: unlabeled data
V: labeled data
(Labeled **pos**itive [relation] or **neg**ative [non-relation])
L: Local classifier
G: Global classifier

BEGIN
  // Initial set, section 3.2
  V = seed set
  Add Non-relations to V [see text]
  Train L, G on V
  REPEAT
    //Co-testing based on L and G, section 3.3
    P = {x ∈ U | G(x) = pos & L(x) = neg}
    N = {x ∈ U | G(x) = neg & L(x) = pos}
    Q = 5 queries selected from P ∪ N, preferring P;
      FOR each q ∈ Q
        //Entity type rules, section 3.5
        IF q violates entity type constraints
          THEN V += <q, neg>
          ELSE V += <q, user-assigned label>
        END IF
      END FOR
    Retrain L, G on V
    //Interleaved self-training, section 3.4
    Self-Train using both L, G
```

```
      to obtain positives and negatives and add to V
    Retrain L, G on V
  END REPEAT
END
```

3.2 Non Relation Approximation

To initiate active learning, we require a small number of seeds (5 in our experiments) for the target relation type. To train the initial model, we also need negative samples. If a small set of negative samples were sufficient, we could ask the user to provide them. However, a small negative set would not be representative of the entire data space, which has far more negative instances than positive ones.[1] As a result, such an initial model gives poor performance; queries in early iterations appear irrelevant to the target relation. Better approximating the negative background by adding a certain number of high-confidence negative samples automatically will give the model the ability to distinguish most negative samples from the very beginning, thus accelerating initial learning.

Random sampling could be used to obtain the negative examples because of the sparsity of positives. However, there is the risk that random sampling may introduce too many false negatives, which is not acceptable for the initial set, even though active learning can deal with a certain degree of noise. To overcome this problem, we train an initial model by incrementally adding more probable non-relations. Since every relation is defined under entity type constraints, we have a subset of the unlabeled data in which the mention pair violates these constraints on the target relation. The instances in this subset are strongly assured not to be target relations if the entity types are hand-labeled, and somewhat more weakly assured if labeled by a NE tagger. By sampling from this subset of non-relations, we safely approximate the non-relation background of the unlabeled data and foster the early learning of the entity type rules. Thus the queries will also be more meaningful to users even at the beginning of the active learning process.

In implementing the sampling, we use the metric of how much of the non-relation subset we have learned instead of specifying a fixed number of instances. We train the model (a basic local feature classifier, the same as that in co-testing, section 3.3) on the labeled instances, apply the classifier to the so-far-unlabeled instances of this subset, and rank the instances by their uncertainty. We repeatedly select the five most uncertain instances, add them to the labeled set, and retrain the model until the model gives mostly correct predictions on classifying the non-relations in this subset. In the experiments, it is tuned to be 99% accurate on non-relations when the model has roughly balanced precision versus recall on target relations. The balanced model will be a better initial model for later active learning. Meanwhile, the way we add non-relations also enforces early learning of entity type constraints.

3.3 Co-testing based query selection

When the initial set is ready, we can start selective sampling and pose queries to improve the model. We use a co-testing method similar to LGCo-Testing (Sun and Grishman 2012), the state-of-the-art active learning algorithm for relation type extension, but give preference to the weaker classifier to get some additional benefit in the early iterations.

LGCo-Testing uses co-testing based on the local view and the global view to select queries. The local classifier uses a rich set of lexical and syntactic features (from both constituent and dependency parses) as well as semantic type information for the arguments. (Zhou et al. 2005; Jiang and Zhai 2007) studied the effectiveness of different features. The global classifier relies on the similarity of relation phrases (the words between the entity mentions), computed based on the shared contexts of these phrases across a large news corpus. The global classifier returns the relation type of the labeled instances to which the unlabeled instance is most similar (a *k*-nearest-neighbor strategy, with *k*=3).[2] The instances on which the two classifiers disagree is the contention set, from which queries are selected. Elements of the contention set are ranked by the KL-divergence, and elements with greater divergence are preferred as queries (because they are likely to be more informative in updating one of the models). Because of the additional knowledge from the global view, this method outperforms other methods in active learning for relation extraction, and thus we choose this method as our query selection function.

[1] The number of non-relation instances (mention pairs that are not the target type) is usually much larger than the number of target relations. In ACE 2004, it's about 25 times larger than the most frequent relation, EMP-ORG.

[2] We closely followed the classifier design in (Sun and Grishman 2012) so that our results would be comparable; the reader is referred there for more details.

While the global view provides valuable additional knowledge, the global classifier, in practice, gives few positive predictions. In principle, when the two classifiers are evenly matched, co-testing should work quite well at selecting informative instances. In this case, their settings favor instances with a positive prediction from the local classifier and a negative prediction from the global classifier, thus influencing the selection of queries. However, in terms of diversity of queries, the global classifier is more capable of discovering unseen instances in the local feature space.

Active learning systems that are based on co-testing may have a similar problem. So we tried to compensate for this by giving preference to the global classifier. In the contention set, the system will first pick as queries instances that the global classifier believes to be positive, and then pick instances that the local classifier predicts to be positive (this may result in selecting queries only from the global classifier in one iteration). The contention set works based on uncertainty. Giving priority to the global classifier is similar to the preference for density in active learning, which usually works better at few labels (Donmez, Carbonell, & Bennett 2007). To save computing time, the selection is only made from the top entropy instances (1000 in our experiments). When there is a substantial amount of annotated data, the local feature model will be able to cover the diversity from the global view. At this point, the contention set will only have examples that the local classifier predicts positive among the top entropy instances, and the priority to the global classifier will not make changes to query selection. We naturally transition to the original uncertainty-based co-testing. This actually gives a kind of mixture of uncertainty-based and density-based methods, which is expected to give better overall performance.

3.4 Interleaving Self-training

At each iteration of co-testing, the contention set from the local and global classifiers will be the candidate set for queries to be given to the user (section 3.3). We would also like to make use of the agreement set – the elements on which the classifiers agree – to further improve the model. We can do so by applying a semi-supervised method, akin to bootstrapping. To integrate this with active learning, we propose to automatically label selected elements of the agreement set at each iteration, thus extending the knowledge directly provided by the user.

We employed the same models as those in active learning for estimating the confidence. In this task, positives are sparse, while negatives are frequent, so we distinguish the strategies for bootstrapping the two classes in the agreement set. For positives in the agreement set, we set a threshold on the local classifier to select sufficiently confident instances in order to avoid errors even when the model is small. We picked the threshold (0.8) based on our observation of early iteration self-training results. The global classifier works as a constraint to avoid semantic drift. (Sun and Grishman 2011) showed that clusters in the global view could be effective constraints in semi-supervised relation extraction. The global classifier, based on the similarity to the few labeled instances, provides a much stricter constraint on predicting an instance to be positive, so no threshold was required. Among those positive agreement instances satisfying the local classifier threshold, we select the most confidently classified instances to label.

In using those instances which both classifiers agree to be negative, we tend to be greedy. In fact, this is again selecting non-relations from unlabeled data, just as in the initial set setting. In the middle of the active learning, the model is more robust to noisy data, and this negative agreement set is also closer to a pure non-relation set. We employ random sampling on this set to emphasize the diversity since we are less concerned about accuracy. To maintain the balance of positives and negatives in the model, we let self-training produce the same number of positives and negatives. To avoid semantic drift away from human annotation, for each class (positive and negative), we limit the number of self-trained instances to be the same as the number of queries (5) at each iteration.

3.5 Entity Type Constraints

Relations are defined within entity type constraints. For instance, the EMP-ORG relation is limited to the types (PER – ORG), (PER – GPE), (ORG – ORG), (ORG – GPE), and (GPE – ORG) in ACE 2004.[3] In supervised learning, this is usually not a big problem.

[3] PER = person, ORG = organization, GPE = geo-political entity: a location with a government.

	30 iterations		stopping point: iterations	at stopping point		supervised learning
	baseline	our system		baseline	our system	
EMP-ORG	58.13	71.52	200	76.81	76.66	75.63
PHYS	34.63	41.16	200	57.85	64.71	67.39
GPE_AFF	18.18	43.01	119	53.69	53.68	63.33
PER-SOC	74.29	68.87	47	65.67	73.13	73.28
ART	25.93	43.33	31	25.45	43.33	74.36
OTHER-AFF	16.67	50.00	22	10.26	50.00	52.17
Overall	37.97	52.98	103	48.29	60.25	67.69

Table 1. Comparison with baseline (F1 score). The overall F score is the direct average of 6 types

Type	# queries in total	#queries that filters apply	Ratio
EMP-ORG	1000	91	9.1%
PHYS	1000	106	10.6%
GPE-AFF	590	84	14.2%
PER-SOC	234	64	27.3%
ART	151	54	35.8%
OTHER-AFF	105	56	53.3%

Table 2. Instances auto-labeled by type constraints

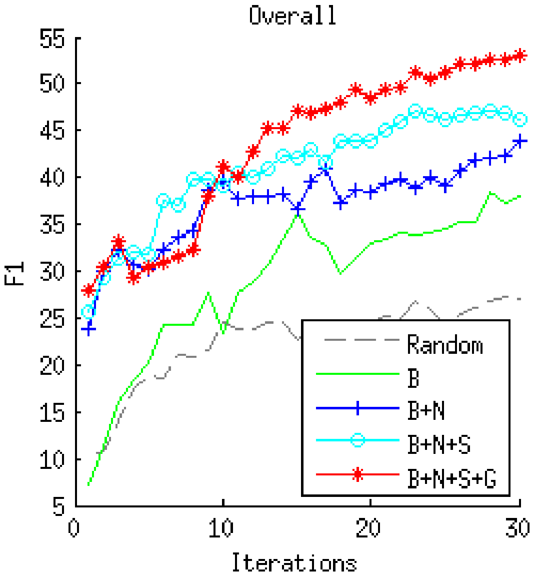

Figure 1. Improvement by different components. B: Baseline, N: Non relations, S: Interleaving Self-Training, G: Preference for the Global View

When the number of instances is large enough, the statistical model will effectively incorporate these entity type constraints as long as entity types are extracted as features. However, in active learning, even with suitable training examples, we will select and present to the user some instances violating these constraints. Applying explicit type filters would save a certain amount of human labeling effort. In practice, this still depends on the quality of the NE tagger. In the experiment section, we show that we can save a certain amount of annotation by using these simple constraints on hand-annotated entities. Since the savings is substantial, especially on some sparse types, it will be still helpful when using an imperfect NE tagger. A similar rule can be constructed to reject candidate relations where the two arguments are co-referential.

4 Experiments

4.1 Experimental settings

We use the ACE 2004 corpus to simulate active learning. We treat each of the relation types in turn as the target type to be learned. We collect all pairs of entity mentions appearing in the same sentence to be the candidates for querying. Our task is to find the target relations and obtain reasonable performance using limited hand-labeled data. We use the original tags in the corpus to answer the queries during the active learning process, which simulates hand-labeling. We take randomly selected 4/5 of the corpus as the sampling space for active learning, and the remaining 1/5 as the test set.

4.2 Evaluation

We compare our work to the pure co-testing based active learning (Sun and Grishman 2012), and show the F1 measure given the same number of iterations (5 queries per iteration). For random selection of target seeds, we use the same random sequence for both the baseline and our framework for fair comparison. In the co-testing framework, the contention set will be empty at some point, which gives the final model of active learning. We report the overall improvement when the system achieves a reasonable performance with limited human annotation (30 iterations) and the final performance (Table 1). The overall result is the average of the F1 measure of all types.

Even though the initial non-relation selection led to early learning of entity type constraints, during the active learning process, there remain queries that could be answered automatically by entity type and co-reference rule filters. The hand-labeling cost could thereby be further reduced (Table 2). For some sparse types, the reduction by these filters is substantial. In practice, this has to deal with noise from the NE tagger, but is still helpful as long as there is a decent NE tagger.

On the whole, our system substantially outperforms the baseline with a small number of labeled examples (150 instances, at the 30th iteration) and also after a relatively large amount has been annotated (the final model)

To show the effectiveness of each component of our framework, we display the overall performance comparison including random sampling, over the first 30 iterations (Figure 1). At this point, most of the six relations have not reached their stopping point, and so the benefits of the individual components are more evident.

The overall F1 score is the direct average of the F1 scores of the six types. Non-relation approximation gives an improvement since auto-labeling a certain number of non-relations saves quite a few queries, and the better initial balance of positive and negative examples also makes the model select more informative queries from the beginning. Self-training boosts the system further as it incorporates more instances (especially positives) automatically. After these, the preference for the global view also gives improvement after 10 iterations. As a trade-off strategy between density and uncertainty, it is common that such methods only outperform the baseline for a certain duration.

With these components and auto-labeling with type constraints (Table 2), we provide a quite reasonable relation extraction system given only 150 labels.[4] With more labels, we can approximate supervised learning. So we can build a relation extraction system quickly when there is no relation annotation in a new corpus. If we need more relations in this new corpus, we can start the framework again, treating previously acquired relations as labeled negative instances of the new target relation. Experiments on this multiple relation type extension also show similar gains over the baseline system using our methods.

5 Conclusion

We present a more practically efficient way to do active learning than a pure co-testing based algorithm. The improvement is most pronounced initially, for small numbers of annotations. We can now achieve reasonable performance for extracting relations with very little annotation. Adding a new relation in an hour now seems within reach.

Each component in the framework is still worth further study. We can consider further efforts to enlarge and balance the initial set from the view of non-relation approximation. We can also try more adaptive semi-supervised algorithms to interleave with co-testing. The quality of the global classifier in the co-testing also remains a constraint, so we will be investigating alternative similarity metrics. While the experiments reported here involve simulated active learning, we are now planning real, human-in-the-loop active learning trials.

Acknowledgment

This research was supported in part by the Intelligence Advanced Research Projects Activity (IARPA) via Air Force Research Laboratory (AFRL) contract number FA8650-10-C-7058 and via Department of Interior National Business Center (DoI/NBC) contract number D11PC20154. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, AFRL, DoI/NBC, or the U.S. Government.

References

Michael Collins. 2002. Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms. *In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP).*

Pinar Donmez and Jaime G. Carbonell and Paul N. Bennett. 2007. Dual strategy active learning. *In Proceedings of the European Conference on Machine Learning (ECML).*

Jing Jiang and ChengXiang Zhai. 2007. A systematic exploration of the feature space for relation extraction. *In Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL).*

[4] Keep in mind that the best systems, trained on thousands of examples, only achieve F scores in the low 70's.

Zornista Kozareva and Eduard Hovy. 2010. Not all seeds are equal: Measuring the quality of text mining seeds. *In Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL).*

Dan Roth and Kevin Small. 2008. Active learning for pipeline models. *In Proceedings of the 23rd National Conference on Artificial Intelligence (AAAI).*

Min Song, Hwanjo Yu, and Wook-Shin Han. 2011. Combining active learning and semi-supervised learning techniques to extract protein interaction sentences. BMC Bioinformatics, **12** (Suppl-12): S4.

Ang Sun and Ralph Grishman. 2012. Active Learning for Relation Type Extension with Local and Global Data Views. *In Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM).*

Ang Sun, Ralph Grishman and Satoshi Sekine. 2011. Semi-supervised Relation Extraction with Large-scale Word Clustering. *In: Proceedings of HLT '11: the 49th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Hans Uszkoreit. 2011. Learning relation extraction grammars with minimal human intervention: strategy, results, insights and plans. *In Proceedings of the 12th* International *Conference on Computational Linguistics and Intelligent Text Processing (CICLing)*.

Vishnu Vyas, Patrick Pantel, Eric Crestan. 2009. Helping Editors Choose Better Seed Sets for Entity Expansion. *In Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM).*

Hong-Tao Zhang, Min-Lie Huang, Xiao-Yan Zhu. 2012. A Unified Active Learning Framework for Biomedical Relation Extraction. *In J. Comput. Sci. Technol.*, **27** (2012), Nr. 6, S. 1302-1313.

Yi Zhang. 2010. Multi-Task Active Learning with Output Constraints. *In Proceedings of the 24th National Conference on Artificial Intelligence (AAAI)*

GuoDong Zhou, Jian Su, Jie Zhang, Min Zhang. 2005. Exploring Various Knowledge in Relation Extraction. *In Proceedings of the 43rd Annual meeting of the Association for Computational Linguistics (ACL).*