Relabeling Distantly Supervised Training Data for Temporal Knowledge Base Population

Suzanne Tamang and Heng Ji
Computer Science Department and Linguistics Department
Graduate Center and Queens College, City University of New York
New York, NY 10016, USA
stamang@gc.cuny.edu, hengji@cs.qc.cuny.edu

Abstract

We enhance a temporal knowledge base population system to improve the quality of distantly supervised training data and identify a minimal feature set for classification. The approach uses multi-class logistic regression to eliminate individual features based on the strength of their association with a temporal label followed by semi-supervised relabeling using a subset of human annotations and lasso regression. As implemented in this work, our technique improves performance and results in notably less computational cost than a parallel system trained on the full feature set.

1 Introduction

Temporal slot filling (TSF) is a special case of Knowledge Base Population (KBP) that seeks to automatically populate temporal attributes or *slots* for people and organizations that occur in national and international newswire sources, and less formal digital publications such as forums or blogs. Typical facts in a knowledge base (KB) contains are attributes for people such as *title*, *residence*, or *spouse* and for organizations, *top employees*, or *members*. We describe work that extends traditional KBP in that not only are relations extracted, but the time for which the relation is valid is also populated, requiring a automated system to construct a timeline for time dependent slot fills.

For many new learning tasks such as TSF, the lack of annotated data presents significant challenges for building classifiers. Distant supervision is a learning paradigm that exploits known relations to extract contexts from a large document collection and automatically labels them accordingly. The distance supervision assumption is that whenever two entities that are known to participate in a relation appear in the same context, this context is likely to express the relation. By extracting many such contexts, different ways of expressing the same relation will be captured and a general model may be abstracted by applying machine learning methods to the annotated data.

Although the distance supervision assumption is generally true, it is considered a weak labeling approach. Recent work in relation extraction has reported challenges using Freebase to distantly supervise training data derived from news documents (Riedel et al., 2010) and TAC's standard slot-filling task (Surdeanu et al., 2010). While extending this framework to TSF, we encounter additional challenges: (1) time normalization results can result in additional errors that proliferate in consequent pipeline steps, (2) Web data is more likely to contradict Freebase facts, and (3) the size of the feature set required to express the rich contexts for a large set of temporal instances can be prohibitively large to learn supervised models efficiently.

To address the challenges associated with noisy, heuristically labeled Web data for training a classifier to detect temporal relations, we improve the accuracy of distantly supervised training data using a semi-supervised relabeling approach, and identify a minimal feature set for classifying temporal instances. The rest of this paper is structured as follows. Section 2 discusses the CUNY TSF system. Section 3 describes our enhancements and how they

Proc. of the Joint Workshop on Automatic Knowledge Base Construction & Web-scale Knowledge Extraction (AKBC-WEKEX), pages 25–30,
NAACL-HLT, Montréal, Canada, June 7-8, 2012. ©2012 Association for Computational Linguistics

were implemented in our experiments. Section 4 presents the experimental results and Section 6 concludes the paper and sketches our future work.

2 Task and System Overview

The temporal KBP slot filling task posed by NIST Text Analysis Conference (TAC) (Ji et al., 2010; Ji and Grisham, 2011) uses a collection of Wikipedia infoboxes as a rudimentary knowledge representation that is gradually populated as new information is extracted from a document collection. This source corpus consists of over one million documents that have been collected from a variety of national and international newswire sources and less formal digital publications. The CUNY TSF system shown in 2 ran several parallel submissions, two that varied only in how the classifier is trained. The methods used to develop the system are described in more detail in previous work (Li et al., 2012).

In order to obtain a large amount of data to train a classifier for labeling temporal instances, we extended a general distance supervision framework for relation extraction (Mintz et al., 2009) and modify the assumption to consider the value of a temporal expression that additionally cooccurs. That is, for a known query, q, attribute, a, and time range, $[t_{begin}, t_{end}]$, sentences in a corpus where q,a, and a temporal expression t co-occur can be automatically labeled with the classes *start, end, hold, range* or *irrelevant* for training purposes using a mapping based on the following heuristic rules and on the value of t:

$$coocur_{q,a,t} = \begin{cases} t = t_{begin}, & start \\ t = t_{end}, & end \\ t_{begin} > t < t_{end}, & hold \\ t = t_{begin} \wedge t_{end}, & range \\ (t < t_{begin}) \vee (t > t_{end}), & irr. \end{cases}$$

As indicated in Figure 2, the system begins with a regular slot filling component to extract slot fills for the given query. Then, document retrieval is performed based on the query and attribute indicated in Freebase. The next step, sentence retrieval, considers the time expression indicated in Freebase, namely that the sentence should include the query, slot fills, as well as candidate time expressions. The remaining processing can be decomposed into two problems: (1) the classification of any temporal expression in the extracted query and slot fill contexts; and (2) temporal aggregation to form a temporal tuple for each query's slot fills. The motivation for this work was to improve classification performance by improving the quality of that data used to generate the classification model.

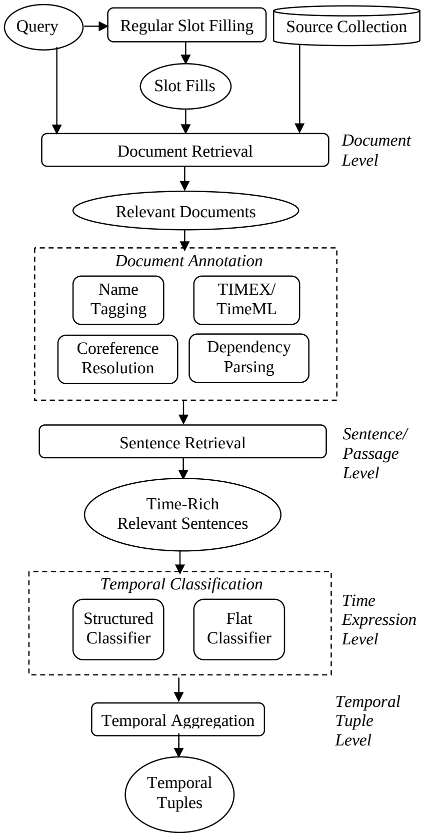

Figure 1: CUNY Temporal KBP System

3 Methods

Table 3 compares the number of temporal relations identified by a human annotator using the TAC KBP corpus with what we were able to retrieve from the Web without human intervention. We can see that our automatic method has obtained much larger training data (more than 40,000 instances). The major advantage of using additional Web data to retrieve candidate temporal instances is the diversity of contexts that can be obtained. For example, expressions captured by this larger data set included common patterns as well less common phrases and

Category	Type	Total	Start	End	Holds	Range	Others
Spouse	Manual	28	10	3	15	0	9
	Automatic	**10,196**	**2,463**	**716**	**1,705**	**182**	**5,130**
Title	Manual	461	69	42	318	2	30
	Automatic	**14,983**	**2,229**	**501**	**7,989**	**275**	**3,989**
Employment	Manual	592	111	67	272	6	146
	Automatic	**17,315**	**3,888**	**965**	**5,833**	**403**	**6,226**
Residence	Manual	91	2	9	79	0	1
	Automatic	**4,168**	**930**	**240**	**727**	**18**	**2,253**

Table 1: Number of human and distantly supervised training instances by dataset

implied information. We used a variety of lexical and syntactic features after document annotation and sentence retrieval to generate a feature set for supervised learning.

3.1 Relabeling

The temporal class labels, *start, end, hold, range* and *irrelevant*, are used to inform the final aggregation that is done for each entity in the KB. In order improve the accuracy and of the training instances and incorporate local context that distance supervision does not capture, we used *self-training*, a semi-supervised learning method that has been used to label data for tasks such as parsing (Mcclosky et al., 2006). Using a small set of human annotations, or *seed* examples, we iteratively labels the partitioned unlabeled set, retaining only the confident labels for retraining the classifier in each round. However, the size of the training dataset resulted in a prohibitively large, sparse feature space. We perform two step in order to generate a more parsimonious classification model that can be used for self-training: (1) *feature elimination* to identify a minimal set of model features, followed by (2) *relabeling* using the reduced feature set and a lasso regression classifier.

Feature elimination: First, for each of the M features in the set $F = \{f_1, ...f_M\}$ extracted from the training data we test the independence of each feature given each class label, inserting only those features that meet a threshold p-value into the minimal feature set F'. Although this approach tests each feature uniquely, many of the features already express conjunctive combinations of tokens.

Self-training: To relabel the instances using the reduced feature set F', we annotated a small set of training data by hand and used lasso (least absolute shrinkage and selection operator) regression, which has the benefit of shrinking the coefficients of features towards zero so that only the subset of features with the strongest effects are incorporated into the classifier (Ng, 2004; Li et al., 2005). The shrinkage parameter ($s > 0$) is tuned using cross-validation. For a collection of N training instances, $D = \{(x_1, y_1), ..., (x_N, y_N)\}$, of d dimensions the lasso coefficients $\hat{\beta}$ are calculated as follows:

$$\hat{\beta}^{lasso} = \text{arg min}_{\beta} \left\{ \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{d} \beta_j x_{ij})^2 \right\}$$

$$\text{subject to: } \Sigma_{j=1}^{d} |\beta_j| \leq s$$

Lasso regression limits the expression of extraneous information and as a result provides additional feature selection properties. The lasso minimizes the residual sum of squares with the constraint that the absolute value of the regression coefficients must be less than a constant, s, that functions as a tuning parameter and is used for shrinkage. When s is large enough, there is no effect on the solution, but when it shrinks it has the effect of reducing some model coefficients close or equal to zero. We used cross-validation to determine the best values for s in our experiments.

In our experiments, we used .005%-.101% of training instances from distant supervision data as the initial labeling seeds for self-training. We used the agreement between classification results for two different values of s, the regularization parameter for the model. As the new data portion is labeled, those retained for retraining are instances for which there is an agreement reached by multiple classifiers.

4 Results

Figure 2 presents the performance of our system on the full TSF task, before and after applying feature selection and re-labeling techniques. The F1 measure for the system that used relabeled training data and the reduced feature space for classification of training instances reported a top F1 measure, slightly improving the overall performance (F-measure from 22.56% to 22.77%). Experimental results on development results have also shown that the F-measure gain on each slot type correlates (.978) with the number of seed instances used in self-training based re-labeling. The most dramatic improvements are obtained for the `per:spouse` slot (7.12% absolute F-Measure gain) which also came the closest to that of human performance.

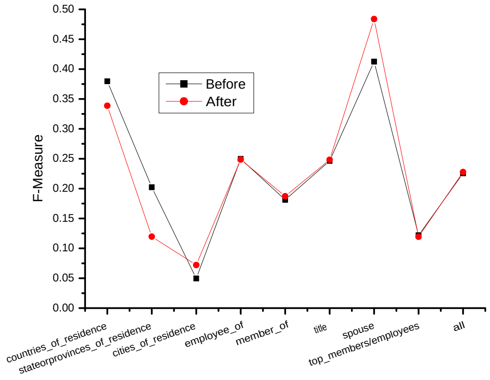

Figure 2: Impact of feature selection and relabeling

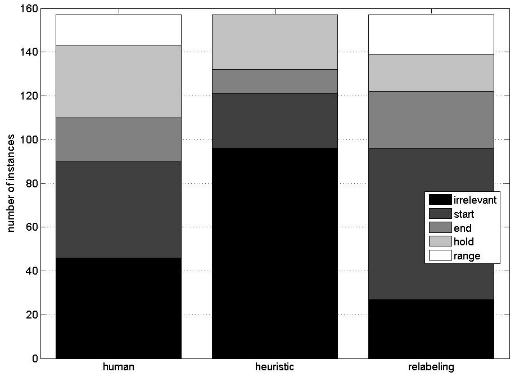

Figure 3: Distribution of class labels

To more closely examine the effects of relabeling on classification, we compared the accuracy of the labels before and after relabeling for the spouse slot type using development data. Since the set of all instances would entail considerable work for a human annotator, we selected 1000 instances at random, eliminating all instances where the labels agreed between the two systems and were left with 83% of all labeled training data. Then, for those instances remaining, a human annotator assigned a *start*, *end*, *hold*, *range* or *irrelevant* label that was used as a gold standard. Figure 3 shows the distribution of labels. Compared with human annotation or after relabeling, the system without relabeling shows a notably higher proportion of irrelevant labels and relatively few range labels. Table 2 further details performance pre-post relabeling, reporting the precision, recall.

5 Discussion

The lack of training data for supervised learning is a bottleneck to improving automated KBP systems and distant supervision offers an attractive approach to expediting the labeling of training data at low cost. Not surprisingly, using heuristics to label temporal instances leads to the introduction of erroneous annotations. Some common causes of error are: coreference results match the wrong named entities in a document, temporal expressions are normalized incorrectly, temporal information with different granularities has to be compared (e.g., we know John married Mary in 1997, but not the exact day and month. Should the time expression September 3, 1997 be labeled *start*?), and information offered by the KB is incorrect or contradictory with information found on the Web documents.

Label	Precision	Recall
Start	.27-.64	.60-.60
End	.10-.55	.29-.50
Hold	.30-.24	.66-.62
Range	0-.64	0-.56

Table 2: Pre-post relabeling preformance

To address these challenges, we develop a simple but effective techniques to relabel temporal instances. Noise is a major obstacle when extending distant supervision to more complex tasks than traditional IE, and our techniques focuses on refining the feature set so that more meaningful features are expressed, and spurious features are removed, or ignored. We perform two steps: using multi-class logistic regression as the basis for eliminating features followed by relabeling with a lasso regression which has additional feature selection properties.

Feature reduction: reasons to perform variable selection include addressing the curse-of-dimensionality, interpretability of the model, and reducing the cost of storing, and processing the predictive variables. We were motivated by the need to provide a more succinct classification model for self-training. Some slots generated over 100,000 features from the training data, and high dimensionality and sparsity was associated with the feature space. Feature reduction with multi-class logistic regression was most dramatic in first development system, which was also the noisiest, averaging 96.2% feature elimination. The classifiers trained on our final system showed an average of 89% feature reduction for the temporal slots, resulting in a more parsimonious classification model.

Relabeling: the procedure described in this work resulted in slightly increased performance on the TSF task. Temporal labels are initially assigned using distant supervision assumptions, which in some cases result in inaccurate labels that could be better informed by local context. For example, the temporal instance below was returned by distant supervision given the query *Jon Voight*, the slot value for the spouse, *Marcheline Bertrand*, and the relevant date range, 1971-1978. Caps are used to show the normalization with the substituted text in brackets:

> "According to former babysitter late mother TARGET ATTRIBUTE [Marcheline Bertrand] virtually abandoned her baby daughter after a painful TARGET DATE [1976] split from husband TARGET_ENTITY [Jon Voight]."

Since the date 1976 is between the range indicated by Freebase it was labeled a target date, and distance supervision heuristics assigned a *hold* label, indicating that the relation was true for 1976, but that it was not the beginning or end. However, the context supports the labeling of this instance more accurately labeled as the *end* of the spouse relation.

Similarly, the following sentence has a date detected that was within the valid range and was also mislabeled, this time as *irrelevant*:

> "TARGET ATTRIBUTE [Shirley] has one daughter, 54, with her TARGET ENTITY [Parker], who she split from in TARGET DATE [1982]."

In this example, a different date was indicated for the *end* of the relation spouse in Freebase. Although supporting text can be used to infer the end of a relation, the simplicity of the distant supervision causes it to fail in this case. Relabeling provided the correct assignment in both of these examples, and it ability to correctly label the instances is likely due to a strong association of the feature 'split_from' with the *end* label.

6 Conclusion

To address the challenges associated with noisy, heuristically labeled Web data for training a classifier to detect the temporal relations, we develop a method with several important characteristics. First, it achieves state-of-the-art performance for TSF, slightly improving on a parallel system that was trained on the full feature set without relabeling. Second, it dramatically reduces the size of the feature space used for labeling temporal instances. Lastly, it can be used to identify which model features are more significant for predicting temporal aspects of a query attribute relation.

Our future work will continue to develop techniques for addressing the challenges posed by extending distant supervision to new types of IE tasks, and the refinement of our techniques. Specifically, it is still unclear how the number of seed instances for semi-supervised relabeling impacts TSF performance and why slot level performance is variable when the number of seed examples is similar. Also, we used a random set of seed examples for self-training and it is possible that learning from certain types of instances may prove more beneficial and that more iterations in the self-training process may continue to improve the accuracy of training labels and overall system performance.

7 Acknowledgements

This work was supported by the U.S. Army Research Laboratory under Cooperative Agreement No. W911NF- 09-2-0053 (NS-CTA), the U.S. NSF CAREER Award under Grant IIS-0953149, the U.S. NSF EAGER Award under Grant No. IIS-1144111, the U.S. DARPA Broad Operational Language Translations program and PSC-CUNY Research Program. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

References

Qi Li and Javier Artiles and Taylor Cassidy and Heng Ji. 2012. Combining Flat and Structured Approaches for Temporal Slot Filling or: How Much to Compress? *Lecture Notes in Computer Science*, 2012.

Heng Ji and Ralph Grishman. 2011. Knowledge Base Population: Successful Approaches and Challenges. *Proc. of ACL2011*, June:1148–1158.

Heng Ji and Ralph Grishman and Hoa Trang Dang. 2011. An Overview of the TAC2011 Knowledge Base Population Track. *Proc. Text Analytics Conference (TAC2011)*, 2011.

Heng Ji and Ralph Grishman and Hoa Trang Dang and Kira Griffitt and Joe Ellis. 2010. An Overview of the TAC2010 Knowledge Base Population Track. Proceedings of the Third Text Analysis Conference, November, 2010.

Mihai Surdeanu and David McClosky and Julie Tibshirani and John Bauer and Angel Chang and Valentin Spitkovsky and Christopher Manning. 2010. A Simple Distant Supervision Approach for the TAC-KBP Slot Filling Task. Proceedings of the Third Text Analysis Conference, November, 2010.

Sebastian Riedel and Limin Yao and Andrew McCallum. 2010. Modeling Relations and Their Mentions without Labeled Text. ECML/PKDD, (3),2010:148–163.

David Mcclosky and Eugene Charniak and Mark Johnson. 2006. Effective self-training for parsing. In Proc. N. American ACL (NAACL), 2006:152–159.

Andrew Y. Ng. 2004. Feature selection, L1 vs. L2 regularization, and rotational invariance. In ICML, 2004.

Fan Li and Yiming Yang and Eric P. Xing. 2006. From Lasso regression to Feature vector machine. NIPS, 2005.

Mike Mintz and Steven Bills and Rion Snow and Daniel Jurafsky 2009. Distant supervision for relation extraction without labeled data. ACL/AFNLP, 2009:1003–1011.