EDL 2016 UTAustin System Description

Nazneen Fatema Rajani and **Raymond J. Mooney**
Department of Computer Science
University of Texas at Austin
nrajani@cs.utexas.edu, mooney@cs.utexas.edu

Abstract

We propose stacking with auxiliary features (SWAF) that combines supervised and unsupervised methods to ensemble multiple systems for the Tri-lingual Entity Discovery and Linking (TEDL) 2016 evaluation. We use the TEDL 2015 systems for training and EDL1 2016 systems for evaluating our algorithm. We perform a post-processing step on the outputs obtained from the classifier so as to aggregate into one final system.

1 Introduction

In 2016, UT Austin was a first time participant in the Tri-lingual Entity Discovery and Linking (TEDL) task of the Text Analysis Conference (TAC) Knowledge Base Population (KBP) evaluation. Every year, many systems participate in the KBP evaluation, in particular, the entity linking task attracts many participants. However, the best performing team has an approximate F1 score of 60 and it remains more or less the same each year. Some of these top systems are high precision while others are high recall and thus overall F1 remained low. In order to get the best of both these type of systems, we were motivated to ensemble them and expectedly obtained good results. The details of this approach is described in our 2015 ACL (**?**) and 2016 EMNLP papers (**?**). Normal stacking trains a classifier to combine the output of multiple systems using as input features the output and confidence of each individual system. In particular, we use features capturing how well the systems agree about the provenance of the entity they link.

We would sometimes like to ensemble systems for which we have no historical performance data. For example, due to privacy, some companies may be unwilling to share their performance on arbitrary training sets. Simple methods such as voting permit "unsupervised" ensembling, and several more sophisticated methods have also been developed for this scenario (**?**). However, such methods fail to exploit supervision for those systems for which we *do* have training data. Therefore, we present an approach that utilizes supervised *and* unsupervised ensembling to exploit the advantages of both. We first use unsupervised ensembling to combine systems without training data, and then use stacking to combine this ensembled system with other systems with available training data.

We demonstrate our approach on the 2016 NIST KBP *Tri-lingual Entity Discovery and Linking* (TEDL) [1]. Figure**??** illustrate our algorithm to combine supervised and unsupervised systems.

2 Overview of the TEDL task

The first step in the TEDL task is to discover all entity mentions in a corpus with English, Spanish and Chinese documents. The entities can be a person, organization, geo-political entity, facility, location (PER/ORG/GPE/FAC/LOC). The extracted mentions are then linked to an existing English KB (a version of FreeBase) entity via its ID. If there is no KB entry for an entity, systems are expected to cluster all the mentions for that entity using a NIL ID. The output for the task is a set of extracted mentions, each with a string, its provenance in the cor-

[1] http://nlp.cs.rpi.edu/kbp/2016/

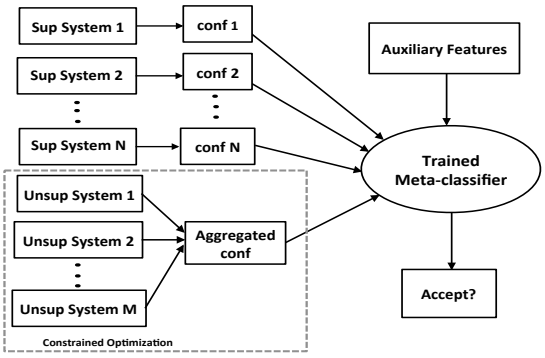

Figure 1: Illustration of our approach to combine supervised and unsupervised ensembles.

pus, and a corresponding KB ID if the system could successfully link the mention, or else a mention cluster with a NIL ID. Systems can also provide a confidence score for each mention.

3 Ensembling Algorithm

Figure **??** illustrates our system which trains a final meta-classifier for combining multiple systems using confidence scores and other auxiliary features depending on the task.

3.1 Supervised Ensembling Approach

For the KBP systems that are common between years, we use the stacking method of **?**) for these shared systems. The meta-classifier makes a binary decision for each distinct output represented as a *key-value* pair. The function of the *key* is to provide a handle for aggregating outputs that are common across systems. For TEDL, the key is the *KB (or NIL) ID* and the value is a *mention*, that is a specific reference to an entity in the text. The top half of Figure **??** illustrates ensembling multiple systems with historical training data using a supervised approach.

3.2 Unsupervised Ensembling Approach

Only 8 of the 16 systems that participated in EDL1 2016 also participated in 2015 TEDL. Therefore, many KBP systems in 2016 were new and did not have past training data. Thus, for improving *recall* and performance in general, it is crucial to use systems without historical training data, which we call unsupervised systems. To achieve this end, we first ensemble such systems using an unsupervised technique. Frequently, the confidence scores provided by systems are not well-calibrated probabilities. So in order to calibrate the confidence scores across unsupervised systems, we use the constrained optimization approach proposed by **?**). The idea is to aggregate the raw confidence values produced by individual KBP systems, to arrive at a single aggregated confidence value for each *key-value* pair. The constraints ensure that the aggregated confidence score is close to the raw score as well as proportional to the agreement among systems on a value for a given key. Thus for a given key, if a system's value is also produced by multiple other systems, it would have a higher score than if it were not produced by any other system. The authors use the inverse ranking of the average precision previously achieved by individual systems as the weights in their algorithm. However since we use this approach for systems with no historical performance data, we use uniform weights across all unsupervised systems for both the tasks.

We use the entity type for the TEDL task to define the constraints on the values. The output from the constrained optimization approach for both tasks is a set of key-values with aggregated confidence scores across all unsupervised systems which go directly into the stacker as shown in the lower half of Figure **??**. Using the aggregation approach as opposed to directly using the raw confidence scores allows the classifier to meaningfully compare confidence scores across multiple systems although they are produced by very diverse systems.

Another unsupervised ensembling method we tried in place of the constrained optimization approach is the Bipartite Graph based Consensus Maximization (BGCM) approach of **?**). BGCM is presented as a way of combining supervised and unsupervised models, so we compare it to our stacking approach to combining supervised and unsupervised systems, as well as an alternative approach to ensembling *just* the unsupervised systems before passing their output to the stacker. BGCM performs an optimization over a bipartite graph of systems and outputs, where the objective function favors the smoothness of the label assignments over the graph, as well as penalizing deviations from the initial labeling provided by supervised models.

Table 1: Results on 2016 Tri-lingual Entity Discovery and Linking (TEDL) (Window 2) using official NIST scorer and CEAF metric

Methodology	Precision	Recall	F1
Combined stacking and constrained optimization using all features	0.789	**0.481**	**0.597**
Combined stacking and constrained optimization using only confidence scores	0.727	0.438	0.546
Stacking only supervised systems using all features	0.551	0.428	0.482
Stacking only supervised systems using only confidence scores	0.597	0.339	0.432
Mixtures of Experts Model (**?**)	**0.848**	0.063	0.116

3.3 Combining Supervised and Unsupervised

We propose a novel approach to combine the aforementioned supervised and unsupervised methods using a stacked meta-classifier as the final arbiter for accepting a given key-value. The outputs from the supervised and unsupervised systems are fed into the stacker in a consistent format such that there is a unique input *key-value* pair. Most KBP teams submit multiple variations of their system. Before ensembling, we first combine multiple runs of the same team into one. We combine the runs of each team into one to ensure diversity of the final ensemble.

The unsupervised method produces aggregated, calibrated confidence scores which go directly into our final meta-classifier. We treat this combination as a single system called the *unsupervised ensemble*. We add the unsupervised ensemble as an additional system to the stacker, thus giving us a total of $N + 1$, that is 9 EDL systems. Once we have extracted the auxiliary features for each of the N supervised systems and the unsupervised ensemble for both years, we train the stacker on 2015 systems, and test on the 2016 EDL1 systems. The unsupervised ensemble for each year is composed of different systems, but hopefully the stacker learns to combine a generic unsupervised ensemble with the supervised systems that are shared across years. This allows the stacker to arbitrate the final correctness of a key-value pair, combining systems for which we have no historical data with systems for which training data *is* available. To learn the meta-classifier, we use an L1-regularized SVM with a linear kernel (**?**) (other classifiers gave similar results).

3.4 Post-processing

Once we obtain the decisions on each key-value pair from the stacker, we perform some final post-processing. For EDL, for each entity mention link that is classified as correct, if the link is a KB cluster ID then we include it in the final output, but if the link is a NIL cluster ID then we keep it aside until all mention links are processed. Thereafter, we resolve the NIL IDs across systems since NIL ID's for each system are unique. We merge NIL clusters across systems into one if there is at least one common entity mention among them.

4 Experimental Results

Table **??** shows the TEDL results. Our full system, which combines supervised and unsupervised ensembling performed the best. An ablation of our best approach that only uses the confidence scores as features was the second best. Our third and fourth best systems are approaches that use SWAF on just the supervised systems. And finally our worst run was a baseline that we wanted to compare against. The approach is called Mixtures of Experts (ME) (**?**) and we submitted that as a run because of its proximity to SWAF in terms of the underlying intuition. In this method, the problem space is partitioned stochastically into a number of sub-spaces and the idea is that the experts or learners are specialized on each subspace. ME uses divide-and-conquer principle to soft switch between learners covering different sub-spaces of the input. This method uses a supervised gating network which can be learnt using Expectation-Maximization (EM). Recollect that the intuition behind using the base features is also very similar.

TEDL provides three different approaches to measuring accuracy: entity discovery, entity linking, and mention CEAF (**?**). CEAF finds the optimal alignment between system and gold standard clusters, then evaluates precision and recall micro-averaged. We obtained similar results on all three metrics and only include CEAF.

5 Conclusion

We presented results on the 2016 TEDL task, showing that a novel stacking-based approach to ensembling both supervised and unsupervised systems is very promising. We found that adding the unsupervised ensemble along with the shared systems specifically increased recall substantially.

Acknowledgment

This research was supported by the DARPA DEFT program under AFRL grant FA8750-13-2-0026.