Pairwise Webpage Coreference Classification using Distant Supervision

S. Shivashankar[1*], Tim Baldwin[2*], Julian Brooke[3*], and Trevor Cohn[4*]

[*]Department of CIS, The University of Melbourne, VIC, Australia

[1]shivashankar@student.unimelb.edu.au
[2,3,4](tbaldwin,julian.brooke,t.cohn)@unimelb.edu.au

ABSTRACT

Multiple URL endpoints exist for an entity on the web. Hence we address the fundamental task of pairwise webpage coreference classification in-order to build coreferent set of endpoints. To strike a balance between unsupervised and supervised methods that require annotated data, we build a positive and unlabeled (PU) learning model where we obtain positive examples using web-search-based distant supervision. We evaluate our proposed approach using SemEval-2007 WePS and ALTA-2016 shared task datasets.

Keywords

Webpage Coreference; End-point Disambiguation; Distant Supervision; PU Learning; Semi-supervised Learning

1. INTRODUCTION

Entity endpoints are URLs which reliably disambiguate named entity mentions on the web[1]. Target entity can be a person, organization, location, event or concept. Entity linking systems typically rely on semantic resources such as Wikipedia, DBPedia to be used as end-points. Though they provide a rich context for entities, these sources do not achieve sufficient recall for many domains. In addition to traditional sources, fortunately many other end-points exist for an entity on the web. For instance, social sources (e.g. facebook.com/*), news aggregation endpoints (e.g. `nytimes.com/topic/person/*`) and organisation directories (e.g. `gtlaw.com/People/*`) present largely untapped sources of valuable entity information. Each source can be treated as a knowledge base (KB) containing entity related unstructured data (webpages). Webpage instances are identified using URL end-points. To build on the automatic approaches for discovering web KBs [1], we focus on the subsequent challenge of end-points reconciliation across KBs. Specifically, similar to ALTA 2016 shared task[1], given two candidate endpoint URLs, the task is to determine whether they refer to the same underlying entity. This can be seen as a precursor to grouping web endpoints into coreferent sets. Web person search (WePS) is a closely related unsupervised task. However, WePS focuses on clustering entity mention pages instead of entity endpoint pages, i.e., the pages can have content related to multiple entities.

Rather than building a completely unsupervised model [1, 2, 5] or a fully supervised model as in the ALTA shared task, we propose an approach to obtain web-search-based distantly supervised positive examples and employ a positive and unlabeled (PU) learning algorithm [3, 4]. The intuition is that, if web-search results with target entity name and context keyword from URL_a as a query, retrieve URL_b in the top search results, then both URLs are more likely to refer to the same entity. But if the top results do not include URL_b, then it may not rule out the possibility completely for the pair to refer to the same entity. E.g., though Wikipedia (W1) and biography.com (B1) pages refer to the singer George Clinton, web search for query 'George Clinton' AND 'P-Funk', where context keyword 'P-Funk' is extracted from `https://en.wikipedia.org/wiki/George_Clinton_(musician)` (W1), does not get `www.biography.com/people/george-clinton-537674` (B1) as one of the top pages.

[1]http://alta.asn.au/events/sharedtask2016/description.html

WWW 2017, April 3–7, 2017, Perth, Australia.
ACM 978-1-4503-4914-7/17/04.
http://dx.doi.org/10.1145/3041021.3054224

2. PROPOSED APPROACH

We assume to have web URL end-point KBs built using automatic approaches [1]. We refer to web URL end-points as URL henceforth. From such KBs, given a training dataset D with URL pairs from different KBs that share the same entity name, our objective is to learn a pair-wise URL classifier. For a URL pair U_1 and U_2, we learn a model $f(x) \rightarrow y$ where the target $y \in \{0, 1\}$ denotes whether the URL pair refers to the same entity ($y = 1$) or not ($y = 0$). x is the feature map to generate pair-wise URL document features ($x = \phi(U_1, U_2)$). Initially all the pairs are unlabeled $D_u{=}D$, positive and negative labeled sets are empty $D_p, D_n = \emptyset$.

2.1. Distant Supervision : We construct web search query for distant supervision as follows: **Q1:** Using target entity name and context information from U_1. Context information includes person name (PER) and organization (ORG) named entities provided by Stanford NER toolkit. Similar to [5], we consider at most 8 context entities from the start of document for querying. The intuition is that context entities in the *title* and document beginning are more likely to be relevant to the target entity. **Q2:** Similar to above, we use target entity name and context information from U_2 for querying.

After querying on DuckDuckGo[2] search engine with the target entity name and each context keyword from U_1 (at most 8 queries), we check if U_2 is present among top K search results (we use $K = 30$). If it is present, then we refer to it as a 'hit'. We also compute results with **Q2** (second set of queries, at most 8 again). We evaluate a simple setup where we assign distant label $\hat{y_{ij}} = 1$ for URL_i and URL_j if there was at least one hit. These positive instances are added to D_p and removed from D_u.

2.2. Labeling Unlabeled Pairs: Since the proportion of positive examples may not be large in all cases, we recover positive and negative examples from D_u. The steps involved are as given below

- Step 1: Choose 'N' samples randomly from D_p, which is denoted as S_p. Remove those points from D_p.
- Step 2: Build a binary classifier θ with D_p and D_u as two classes (D_p vs D_u). We use SVM binary classifier with linear kernel.
- Step 3: Compute average (probabilistic) score of instances in S_p given by θ (using Platt scaling), $\mu_p = \frac{1}{|S_p|} \sum_{i:S_p} p(x_i = 1)$. We use it as the threshold for labeling data-points in D_u, which gives D_u^l. If $p(x_u = 1) > \mu_p$ then label it as positive, negative otherwise.

We see this as a variant approach built on [3] and [4]. In [4] a threshold is derived from S_p to only recover negative examples from D_u. In [3] uses average decision score of positive examples given by θ to scale the threshold for test data finally. Above mentioned step-wise procedure gives a dataset with positive and negative examples ($D_p \cup D_u^l$) that can be used with any standard binary classifier. We use SVM with linear kernel and we refer to it as **DP-SVM** in Table 1, since it uses propagated distant labels.

3. EXPERIMENTS

We use SemEval-2007 WePS development[3] set and ALTA-2016 shared task datasets. We constructed a balanced 1000 URL pair dataset from webpages belonging to 49 persons. We only included pairs sharing the same person name from the webpages that still exist. Additionally, we considered end-point pages only (filtered using features from [1]). If the pair refers to same entity, then it is a positive example, negative otherwise. We used a random split of 70-30 for training and testing. The ALTA dataset is a balanced set which contains 400 pairs of URLs that can refer to any entity – person, organization, location, etc. We used 300 pairs for training and retained the private test set of 100 pairs from competition for testing. Note that we use manual annotations provided in the dataset only for testing.

We extract standard set of features proposed by best performing systems in the respective competitions. Briefly, the 8 features created on WePS dataset include: unigram based cosine similarity, n-gram based cosine similarity, named entities phrase based cosine similarity (obtained using Stanford NER), title unigram Jaccard Coefficient score, URL character 4-gram Jaccard Coefficient, webpage contents' vector dot product similarity obtained using average Word2Vec representation, document length difference and URL path length difference. We extract the same set of features on ALTA dataset, in addition we also compute semantic similarity[4] and machine translation metrics[5] (BLEU, METEOR and TER) between pairs of Bing search snippet text that was provided in the dataset (in total 12 features).

Using distant supervision (section 2.1) we got around 10% positive examples on WePS dataset and around 25% on ALTA dataset. With D_p as positive and D_u as negative examples, we use biased SVM (BSVM) which has different costs for positive and negative classes as a simple baseline. PU Learning baselines with D_p and D_u include: Spy-SVM with recommended parameters in[4] and SPUL [3]. All SVM based models use a linear kernel. We also evaluate hierarchical clustering based unsupervised approach (referred to as HC) which is shown to perform well for WePS task [2]. We set number of clusters as two and use brute-force technique that minimizes overall error to assign clusters to classes. Parameters for SVM (C) and biased-SVM (C and class weights) are chosen by cross-validation. For our proposed method (DP-SVM), that uses labels from methods given in Section 2.1 and 2.2, we provide average scores across five runs with five random positive examples as S_p. We compute average F-measure on test-set and the results are given in Table 1, best scores are given in bold. Distant supervision based model which uses labels from procedure given in Section 2.1 (BSVM) performs better than unsupervised approach (HC). SPUL is sensitive to ratio of positive examples (D_p vs D_u), it performs better on ALTA dataset compared to WePS dataset. Overall, our proposed approach (**DP-SVM**) performs better than other competing methods. We evaluated the upper bound performance of supervised SVM model built with manually annotated labels. The average F-measure was around 5% greater than our proposed approach on both the datasets.

Table1: Micro-Average F-measure

Dataset	BSVM	Spy-SVM	SPUL	HC	DP-SVM
WePS	0.472	0.63	0.516	0.408	**0.653**
ALTA	0.5	0.54	0.587	0.481	**0.608**

4. CONCLUSIONS

We have evaluated two key contributions: (a) use of distant supervision for web URL endpoints coreference classification task (b) effectiveness of using PU Learning for this setup. To the best of our knowledge this is the first attempt to leverage distant supervision in conjunction with PU Learning.

5. REFERENCES

[1] A. Chisholm, W. Radford, and B. Hachey. Discovering entity knowledge bases on the web. In *Workshop on Automated Knowledge Base Construction*, 2016.

[2] A. D. Delgado, R. Martínez, V. Víctor Fresno, and S. Montalvo. A data driven approach for person name disambiguation in web search results. In *COLING*, 2014.

[3] C. Elkan and K. Noto. Learning classifiers from only positive and unlabeled data. In *SIGKDD*, 2008.

[4] B. Liu, W. S. Lee, and X. Li. Partially supervised classification of text documents. In *ICML*, 2002.

[5] R. Nuray-Turan, Z. Chen, D. V. Kalashnikov, and S. Mehrotra. Exploiting web querying for web people search in weps2. In *Web People Search Evaluation Workshop, WWW*, 2009.

[2] www.duckduckgo.com

[3] http://nlp.uned.es/weps/weps-1

[4] https://dandelion.eu/semantic-text/text-similarity-demo/

[5] https://github.com/jhclark/multeval