AAAI Technical Report WS-12-02
Real-Time Analysis and Mining of Social Streams

Unsupervised Real-Time Company Name Disambiguation in Twitter

Agustín D. Delgado Muñoz, Raquel Martínez Unanue,
Alberto Pérez García-Plaza, Víctor Fresno
NLP&IR Group of UNED University,
Madrid, Spain
{agustin.delgado,raquel,alpgarcia,vfresno}@lsi.uned.es

Abstract

This paper presents a new approach to disambiguate company names in the Twitter social network. We have focused on making lighter the processing of comparing company profiles with tweets in order to obtain a competitive real-time system. With this aim, we only use the home page of each company as information source to create a unique profile. On the other hand, we compute the similarity of a tweet in connection to a profile by comparing the content of the tweet with the profile. Both steps do not use any other external information source and all the process is developed in an unsupervised way. We have tested our application with the test WePS-3 CLEF ORM corpus obtaining encouraging results.

Introduction

One interesting application of named entity (NE) disambiguation is monitoring the online reputation of companies. An intermediate step for this task is to retrieve real information of companies given by users on Internet. In general, company names can usually be ambiguous, for example: the query "Amadeus" may refer to the company "Amadeus IT" or to the musician Wolfgang Amadeus Mozart; the query "Apple" may refer to the IT company Apple Inc., to the fruit, or to any person whose surname is Apple. We have focused our work on the successful Twitter[1] microblogging social network. In this scenario, this task becomes more difficult due to the limited number of characters of Twitter messages–called "tweets"–, which implies to have a small context to perform the disambiguation process. In 2010, WePS-3[2] CLEF evaluation campaign included the Online Reputation Management (ORM) task which motivated our work. The main goal of ORM is, given a set of tweets including the same company name, to build a system capable of deciding which tweets are related to the company and which of them are unrelated. Some approaches were previously presented for solving this task. They used a wide range of information for disambiguating that even includes, in some cases, manually produced terms. Some other works used predefined thresholds for classifying tweets as belonging or not to a certain company.

In this paper we present our approach for this task. Our system is totally automatic, unsupervised and uses only two information sources–the content of company home pages and the tweets themselves–, with the intention of improving its computational complexity in order to use it in real-time applications.

This paper is organizated as follows: in Related Work we comment several works around this task presented in the WePS-3 evaluation campaign. Then, we describe our approach, consisting of two stages to achieve the disambiguation, in System Description Section. First, we deal with the problem of representing the information of a company in a unique profile in Profile Representation Subsection. Next, in Tweet Disambiguation Subsection, we describe our tweet representation approach and our unsupervised method to decide whether a tweet is related to a company profile or not. The experiments carried out are detailed next and, lastly, we present our conclusions and future lines of research.

Related Work

WePS (Web People Search) is a competitive evaluation campaign that proposes tasks around clustering, attribute extraction and resolution of disambiguation on the Web data to research groups. In particular, WePS-3 included an Online Reputation Management (ORM) task focused on the resolution of ambiguity for company names. Five research groups participated in this exercise and the best results were obtained by LSIR (Yerva, Miklós, and Aberer 2010) and IT-UTC (Yoshida et al. 2010) teams reaching an accuracy of 0.83 and 0.75 respectively. The results of all the participant systems can be seen in (Amigó et al. 2010) . The best system, LSIR, extracts terms of each company from external sources as well as from a set of manually fixed terms. It builds a set of six company profiles, one corresponding to each source: company home page, metadata HTML, category/Wordnet, GoogleSet, UserFeedBack Positive and UserFeedBack Negative. These profiles are compared with the corresponding tweets for extracting features that give information to label them as related or unrelated by means of a SVM classifier. Therefore, this system is supervised, needs training data and makes use of external and manual informa-

[1] www.twitter.com

[2] http://nlp.uned.es/weps/

tion. The second ranked system, ITC-UT, uses a two-stage algorithm. In a first step, they categorize queries predicting the class of each company using a Naive Bayes classifier with six binary features (for example, is the query an acronym?). They use thresholds manually established by looking at the training data results for this categorization. The second step consists in categorizing the tweets using a set of heuristics. The SINAI system (García-Cumbreras et al. 2010) uses heuristics based on the occurrence of NEs and external sources as Wikipedia or DBPedia and the company home page. They obtained the third position in the competition ranking. The UVA system (Tsagkias and Balog 2010) uses information based on the language of the set of tweets ignoring external sources. Finally, KALMAR (Kalmar 2010) system uses a bootstrapping method from the terms occurring in the company home page.

We think LSIR and IT-UTC solutions could lead to a higher cost since they use a SVM and Naive Bayes classifiers respectively. SINAI system only takes into account the occurrence of NEs. We think this is a good assumption but it is not always enough. For example, taking the query "Johnnie Walker", some relevant terms are "whisky" or "drink" and they are not NEs. Furthermore, users in Twitter usually write their messages in an informal way and they do not usually refer to other NEs besides the company names. UVA system pretends to see the performance of a system without external information, using only the text of the messages, but they get low accuracy. Finally, KALMAR manually collects the external information and also get low accuracy.

System Description

Our purpose is to build a system able to classify tweets as related or unrelated to a certain company. For this task we want to use as little information as possible in order to use it as a real-time application. To do this, we want to compare tweets with companies in a lighter way than other systems do. We understand company name disambiguation as a two-stage process. Firstly, we compute offline a representation of the company called *company profile*. Secondly, we apply a real-time tweet disambiguation method for deciding whether a tweet is related to a company. We base our approach on the idea of representing the information of a company by means of a unique profile. This profile consists of a bag of stemmed words with their associated weights and it is obtained using a representation based on a fuzzy combination of criteria that we explain next. We gather these terms only via the companies' home pages given in the dataset of WEPS-3 for the ORM task. Then, we apply a tweet disambiguation method by computing a comparison function between two bags of stemmed words: the company profile and the tweet content. Next we use an unsupervised threshold for categorizing each tweet as related or unrelated to the company.

Profile Representation

The first problem our system tackles consists in building a profile for each company – we decided to use a single profile approach – by combining different criteria. Our starting point are the home pages corresponding to those companies. This way, our problem can be seen as a problem of web page representation, where the main goal is to extract the most important information from these web pages to represent each company.

For a human reader, title and emphasized words in a text document have a bigger role than the rest of the words in understanding its main topic. Moreover, the beginning and the end of the body text usually contain overviews, summaries or conclusions with essential vocabulary. The goal of *efcc* (Paukkeri et al. 2012) is to define the importance level of each word in a document by using a set of heuristic criteria: word frequency counts in titles, emphasized text segments, in the beginning and the end of the document, and in the whole document. As titles and other special texts are encoded with HTML tags, a subset of those tags are used in *efcc* in order to collect "the most important" words in a document.

The fuzzy system is built over the concept of linguistic variable. Each variable describes the membership degree of a word to a particular class. The variables are defined by human experts. The fuzzy system knowledge base is defined by a set of IF-THEN rules that combine the variables. The aim of the rules is to combine one or more input fuzzy sets (antecedents) and to associate them with an output fuzzy set (consequent). Once the consequents of each rule have been calculated, and after an aggregation stage, the final set is obtained.

The *efcc* IF-THEN rules are based on the following ideas: (1) If a word appears in the title or the word is emphasized, that word should also appear in one of the other criteria in order to be considered important. (2) Words appearing in the beginning or at the end of a document may be more important than the other words, because documents usually contain overviews and summaries in order to attract the interest of the reader. (3) It is possible that there are no emphasized words in a document. (4) It is possible that a document does not have a title, or that the title does not contain important words. (5) If the previous criteria were not able to choose the most important words, the frequency counts may help to find them.

The inference engine that evaluates the fired rules is based on the *center of mass* (COM) algorithm that weights the output of every fired rule, taking into account the truth degree of its antecedent. The output is a linguistic label (e.g., 'Low', 'Medium', 'Very High') with an associated number related to the importance of a word in the document, and it is calculated by scaling the membership functions by product and combining them by summation. These kind of systems are called additive (Kosko 1998) and their main advantage is the efficiency of the computation.

The rule base for *efcc* is shown in Table 1. Each row has the values of different criteria and the resulting output, called 'Importance'. The main idea is to have two sets of rules: one for frequencies and the other for the rest of the criteria, in such a way that we have always at least one rule of each set fired by the system which will combine the outputs. Input frequencies are normalized with criterion maximum frequency in the document, since we want to grant independence to the rules regarding the document size.

Table 1: Rule base for *efcc*. All inputs are related to term frequencies in different criteria.

IF	Title	AND	Frequency	AND	Emphasis	AND	Global-Position	THEN	Importance
	High				High			⇒	Very High
	High				Medium		Preferential	⇒	High
	High				Medium		Standard	⇒	Medium
	High				Low		Preferential	⇒	Medium
	High				Low		Standard	⇒	Low
	Low				High		Preferential	⇒	High
	Low				High		Standard	⇒	Medium
	Low				Medium		Preferential	⇒	Medium
	Low				Medium		Standard	⇒	Low
	Low				Low		Preferential	⇒	Low
	Low				Low		Standard	⇒	No
			High					⇒	Very High
			Medium					⇒	Medium
			Low					⇒	No

Table 2: Rule base for the auxiliary fuzzy system.

IF	position	THEN	global-position
	Introduction		Preferential
	Body		Standard
	Conclusion		Preferential

The last criterion, word global position, is obtained by means of an auxiliary fuzzy system (Table 2), which takes as inputs all the word positions in the document and returns the global position.

The *efcc* approach makes it possible to combine different criteria to represent company profiles. Furthermore, each profile is represented by means of a single vector within the Vector Space Model (VSM), where each vector component corresponds to the importance of a concrete document term.

Tweets Disambiguation

The tweets in the WePS CLEF ORM corpus are included in JSON files. We extract the text and the ID of those tweets. We later remove stopwords and stem each word using the Porter's algorithm. In this way, our representation of a tweet is a bag of stemmed words.

Given the profile representation of each company that we created in the previous stage, next we compute how related is a tweet with respect to a company using the same function than LSIR system: given a tweet seen as a bag of stemmed words $T = \{wt_1, wt_2, ..., wt_n\}$ and a profile $P = \{(wp_1, w_1), (wp_2, w_2), ..., (wp_m, w_m)\}$ where $\{wp_1, wp_2, ..., wp_m\}$ are the terms and $\{w_1, w_2, ..., w_m\}$ their weights, the following function computes the weight of the tweet T with respect to the company profile P:

$$F(T, P) = \sum_q w_q \tag{1}$$

where q is such that:

$$wp_q \in T \cap \{wp_1, wp_2, ..., wp_m\} \tag{2}$$

It is very frequent to get the company name as a profile term with a high associated weight. As it always appears in all tweets and it is the object of the disambiguation, we delete the company name from the list of profile terms. In this way, the company name does not sum its weight.

We classify the tweets using an unsupervised threshold that we compute automatically. We assume the hypothesis that the profile terms with high weight are enough to determinate the relation between a tweet and a company. Whereas the profile terms with low weight usually are common words that do not solve the ambiguity by themselves, except if a single tweet contains more than one of those terms. Our idea is to define an upper threshold, lower than the weight of the most relevant terms, but higher than most of the profile terms. We compute our unsupervised threshold as follows: we obtain the arithmetic mean of the profile weights and later we compute the arithmetic mean between that number and the maximum profile weight.

$$Avg = \frac{\sum_{i=1}^{m} w_i}{n} \tag{3}$$

$$\gamma = \frac{Avg + \max_i\{w_i\}}{2} \tag{4}$$

To decide whether a tweet is related or unrelated with respect to a certain company, we use the following criteria: if $F(T, P) > \gamma$ we say that the tweet T is related to the company with profile P, if $F(T, P) < \gamma$ we say that T is unrelated to the company and, if $F(T, P) = \gamma$ then we cannot classify that tweet and we say that T is unknown.

Experiments

The WePS-3 CLEF ORM corpus includes 3 datasets: the first one is a trial dataset consisting in 24 companies (18

Table 3: Metric results for the evaluated systems sorted in descending order by Accuracy values. We use bold font to highlight the best result for each metric. Our system is surrounded by a box.

System	Accuracy	Precision (+)	Recall (+)	F (+)	Precision (-)	Recall (-)	F (-)
LSIR	**0.83**	0.71	0.74	**0.63**	**0.84**	0.52	0.56
ITC-UT	0.75	0.75	0.54	0.49	0.74	0.6	0.57
Proposal system	0.69	0.6	0.46	0.44	0.63	**0.80**	**0.65**
SINAI	0.63	**0.84**	0.37	0.29	0.68	0.71	0.53
UVA	0.56	0.47	0.41	0.36	0.6	0.64	0.55
KALMAR	0.46	0.48	**0.75**	0.47	0.65	0.25	0.28

from English speaking contries and 6 from Spanish speaking countries) with 100 tweet for each organization. The second one is a training dataset including 49 company names and around 700 tweets for each company. The last one is the test dataset which includes 47 company and around 500 tweets for each company. In all these datasets are included the URLs of the companies and a set of tweets manually labeled as related or unrelated. Accuracy (ratio of correctly classified tweets) was the metric used to rank the systems. We evaluate our approach with the WePS-3 CLEF ORM test corpus. Table 3 shows the results of our system together with the results of other systems. In that table we can also see other metrics like precision, recall, and F-measure for related tweets (+) and unrelated tweets(-), all of them, defined in (Amigó et al. 2010).

Our system gets an accuracy of 0.69 only below than LSIR and ITC-UT systems. Our method tends to label tweets as unrelated because our threshold is high. Because of this, we obtained high values for recall and F-measure with negative examples. We have seen that a lower threshold tends to increase the false positive cases losing precision and accuracy for both, positive and negative examples. Our system improves the top system LSIR results when they only use the same information than us, the home page profile, getting 0.66 of accuracy as shown in (Yerva, Miklós, and Aberer 2010). We believe we have a good trade-off between our profile representation method and comparison function to achieve good results with cheap computational cost.

Conclusions and Future Work

We have implemented an automatic and unsupervised system for real-time classification of tweets as related/unrelated to a certain company name. Our approach relies on creating a single profile for each company from the content of their home pages. We use information coming from different home page features including title, emphasis, word positions and word frequency. On the basis of this information, we employ a nonlinear combination of criteria based on heuristic knowledge. As a result of this process, we obtain a profile for a given company: a single vector within the VSM. This light representation for company profiles allows to disambiguate tweets in real time by comparing those tweets against the corresponding company profile. This comparison process is computationally cheap, as we just work with two vectors using the method described in Subsection Tweet Disambiguation, having linear complexity. We tested our system using the WePS-3 CLEF ORM corpus obtaining an accuracy of 0.69 only lower than two systems: LSIR and IT-UTC. Our system does not need manual information as LSIR does, and it does not have to learn any threshold using training data as IT-UTC does. We do not employ any classifier as SVM or Naive Bayes. We have proposed an automatic way of computing a threshold for disambiguating tweets. Our experimental results are promising taking into account that we use only one information source–company home pages–for representing the companies. As future work, we will try to extract terms from Wikipedia pages corresponding to each company to extend our current company profile representation. Another field to explore is the possible modification over the fuzzy system rules.

Acknowledgment

The authors would like to thank the financial support for this research to MA2VICMR consortium (S2009/TIC-1542, http://www.mavir.net), a network of excellence funded by the Madrid Regional Government and the Spanish research project Holopedia funded by the Ministerio de Ciencia e Innovación under grant TIN2010-21128-C02.

References

Amigó, E.; Artiles, J.; Gonzalo, J.; Spina, D.; Liu, B.; and Corujo, A. 2010. Weps3 evaluation campaign: Overview of the on-line reputation management task. *CLEF*.

García-Cumbreras, M. A.; García-Vega, M.; Martínez-Santiago, F.; and Peréa-Ortega, J. M. 2010. Sinai at weps-3: Online reputation management. *CLEF*.

Kalmar, P. 2010. Bootstrapping websites for classification of organization names on twitter. *CLEF*.

Kosko, B. 1998. Global stability of generalized additive fuzzy systems. *IEEE Transactions on Systems, Man, and Cybernetics - C* 28:441–452.

Paukkeri, M.-S.; García-Plaza, A. P.; Fresno, V.; Unanue, R. M.; and Honkela, T. 2012. Learning a taxonomy from a set of text documents. *Applied Soft Computing* 12(3):1138 – 1148.

Tsagkias, M., and Balog, K. 2010. The university of amsterdam at weps3. *CLEF*.

Yerva, S. R.; Miklós, Z.; and Aberer, K. 2010. It was easy, when apples and blackberries were only fruits. *CLEF*.

Yoshida, M.; Matsushima, S.; Ono, S.; Sato, I.; and Nakagawa, H. 2010. Itc-ut: Tweet categorization by query categorization for on-line reputation management. *CLEF*.