# Entity Disambiguation in Tweets leveraging User Social Profiles

Surender Reddy Yerva, Michele Catasta, Gianluca Demartini, Karl Aberer
{surenderreddy.yerva@epfl.ch, michele.catasta@epfl.ch, demartini@exascale.info, karl.aberer@epfl.ch}

***Abstract*—Pervasive web and social networks are becoming part of everyone's life. Users through their activities on these networks are leaving traces of their expertise, interests and personalities. With the advances in Web mining and user modeling techniques it is possible to leverage the user social network activity history to extract the semantics of user-generated content. In this work we explore various techniques for constructing user profiles based on the content they publish on social networks. We further show that one of the advantages of maintaining social network user profiles is to provide the context for better understanding of microposts. We propose and experimentally evaluate different approaches for entity disambiguation in social networks based on syntactic and semantic features on top of two different social networks: a general-interest network (i.e., Twitter) and a domain-specific network (i.e., StackOverflow). We demonstrate how disambiguation accuracy increases when considering enriched user profiles integrating content from both social networks.**

## I. Introduction

With the advent of Web 2.0, people being part of many social networks express themselves on various on-line platforms. A part of the users personality is latent among the different actions performed on social networks that they use. Given such user-generated data, it is possible to infer some components of user's personality and accordingly construct user profiles.

For example, an expert in map-reduce and cloud technologies would publish content more often about these technologies as compared to the average user. It could be through writing blog posts, or through microposts on Twitter, or through answering questions on Community QA (CQA) websites. In some cases, user generated content carries clues of user expertise and interests. Thus it becomes, in general, possible to infer expertise model from the user-generated content. Accurately constructing user profiles from their generated content is useful in many scenarios, such as:

**Snapshot View**: the user profiles we construct provide a summarized view of the user presence on on-line social networks.

**Enhanced User Tagging on a Social Network**: the user can be suggested with new tags (learned, e.g., from his Twitter network) which describe himself on a new social network (e.g., on StackOverflow[1]).

**Enhanced Recommendation**: better recommendation engines can be built which can make recommendations based on the constructed user profile.

[1]http://stackoverflow.com/

**Information Filtering**: The generated user profile can be used to filter relevant information from a stream of Web content based on the user interests.

In the current paper, we show a number of techniques of constructing user social profiles, we discuss their merits and demerits, and experimentally compare each of the techniques for constructing such profiles on the task of entity disambiguation. The different user profiling techniques we propose are:

**Term Popularity**: this method reports the top words of a user based on the observed frequencies of the different terms in the user generated content.

**TF-IDF**: in this method we consider those top words after sorting them based on their TF-IDF score.

**Semantic**: we make use of semantic techniques to extract concepts and categories from user-generated documents.

**Topic Modeling**: the top topics related to the user-generated content extracted using Latent Dirichlet allocation (LDA) topic modeling.

**Labeled-LDA Topic Modeling**: summary of user-generated content in terms of labeled tags and words obtained by means of labelled LDA (LLDA) [21].

In this paper we focus on using social network user profiles for effectively addressing the task of disambiguating entity mentions in social network content (i.e., understanding whether the mention of an entity like 'apple' refers to the fruit or to the company)[3] by exploiting the content generated by users on other social networks. We explore how user profiles could be useful for extracting knowledge from data. Some examples of extracting knowledge from an unstructured data, like text documents, include named entity extraction [17], entity reference disambiguation [5], sentiment extraction [16], linguistic tasks [18], etc. Various semantic and knowledge engineering techniques rely on the context for automatic meaning inference from a text [6]. Such techniques are successful for longer documents, as they provide enough context for the proposed tools. However, they can not be directly applied to short texts created within the social network platforms.

Microposts are short texts posted by users on various social networks. Being short texts, microposts usually do not contain enough contextual information for making sense of them. While it would be difficult to develop new techniques that do not need such contextual information, we instead propose to use existing disambiguation techniques and rather to enhance the context of microposts by looking at user activity over other on-line social networks.

The proposed method for entity disambiguation in microposts is based on standard text classification using features extracted from the social network activities of the users. We experimentally compare the effectiveness of the proposed approach by disambiguating entity mentions in Tweets using as background information the user generated content on StackOverflow, a technical CQA system for computer programming topics.

Experimental results show that the classifiers built on top of enriched user profiles significantly outperform the classifiers built on top of the basic user profiles by at least $11\%$. The most effective approach is obtained using frequency-based and LLDA-based user profiles. By combining profiles constructed for a user over different social networks, it is possible to obtain a global social profile for the user which outperforms the other techniques in the tweet disambiguation task.

The rest of the paper is organized as follows. Section II formally presents the user entity profile construction problem and the microposts disambiguation problem. Section III presents the overview of our approach. While Section III-A discusses a number of techniques for constructing user profiles, Section III-B discusses how to solve the microposts classification task. Section IV provides a detailed description of the datasets and the experimental evaluation of the proposed user models. Section V summarizes the related work. Section VI concludes the paper.

## II. Problem Statement

In this section we formulate the two tasks we are addressing in this paper: the creation of a user profile from the user's social network content and the task of classifying a Twitter message based on its relatedness to a company entity.

*Task 1:* A user $u_i$ publishes a set of micro-posts (ex: *tweets*,*comments*) on a social network. We group such microposts of a user together as a document $D_i = \{m_1, m_2, \ldots, m_n\}$. We model the user profile $U_i$, of user $u_i$, as a bag of weighted set of keywords i.e. $U_i$ = $Set\{wrd_k : wt_k\}$ with weights being normalized. These set of keywords could represent the topics or concepts that are most likely to occur in the user's microposts. We define $Corpus$ as the group of documents related to the various users of the system: $Corpus = \{D_1, D_2, \ldots, D_m\}$. We define the topic extraction as a function $f$: $D_i$ x $Corpus \Rightarrow U_i$. The techniques we considered are discussed in Section III-A.

*Task 2:* Given a set of Twitter messages $\Gamma = \{T_1, \ldots, T_n\}$ containing an ambiguous company name (e.g., *apple*, *orange*), we want to classify whether the message is related to a given company entity $C$ or not. We say that the message $T_k$, created by user $u_i$, is related to the company $C$, $related(T_k, C)$, if and only if the Twitter message refers to the company. We also use the term that a tweet belongs to a company, by which we mean the same. We assume that some basic further information is available as input, such as the URL of the company $url(C)$ and the language of the Web page.

The tweet messages are modeled as a bag of words. Each tweet is preprocessed through following steps: we remove stopwords, emoticons, and Twitter specific stop-words (such as, for example, RT,@username), we store a stemmed (using the *Porter stemmer*[2]) version of keywords (unigrams and bigrams). Formally we have: $T_k = Set\{wrd_j\}$.

The company entity $C$ is modeled as a set of weighted keywords. The company entity: $C$ = $Set\{wrd_k : wt_k\}$, with $wt_k \geq 0$ for positive evidence keywords (i.e. those words which suggest that the message should be related to the company) and $wt_k < 0$ for negative evidence keywords. We discuss the classification of tweet messages belonging to a company entity in Section III-B.

## III. System Overview

Users are typically present on several on-line social networks. They publish microposts on their Twitter stream, comments and posts on Facebook, address questions and answers on CQA sites like StackOverflow, express their interests through Facebook likes and Google +1s, etc. All the content users post, his activities on the web, and his social network interaction data can be tremendous value for automatically constructing a part of the user personality.

In this section, first we present a number of techniques for constructing user entity profiles. In the second part we address the problem of classifying a micropost (tweet) based on whether it is related to a company entity or not.

### A. User Entity Profile Techniques

*1) Frequency and TF-IDF based Topics:* TF-IDF is often used in information retrieval and text mining for weighting document terms. A term is considered as important to a document if it appears more often in the document itself and tends to appear in fewer documents in the corpus. Term-frequency (TF) captures how often a particular word appears in a document, while inverse-document-frequency (IDF) captures how rare a particular term is in the document corpus.

$$tf(w_i, D) = \frac{freq(w_i, D)}{max\{freq(w_k, D); \text{for word } w_k \in D\}} \quad (1)$$

$$idf(w_i, D) = log \frac{|CorpusSize|}{\text{Number of docs containing the } w_i} \quad (2)$$

$$tf\text{-}idf(w_i, D) = tf(w_i, D) * idf(w_i, D) \quad (3)$$

A user entity profile $U_i^{tf}$ is constructed based on the TF metric. We choose the top-K (with $K$ ranging from 10 to 150) terms with the highest TF score (eqn. 1) to be present in the user entity profile. Similarly we construct another user entity profile $U_i^{tf^{idf}}$ based on the TF-IDF metric (eqn. 3). The top-K terms with the highest TF-IDF score are stored in this user profile.

The frequency based user profile $U_i^{tf}$ is independent of the corpus, as it only depends on the current user-generated document. Such property allows a relatively efficient construction this profile. However, when a user tends to publish tweets related to various topics, for example: *technology*, *sports* and *politics*), and one of such topics is predominant, then the frequency based profile fails to capture the diversity in the different topics the user is writing about.

[2]http://tartarus.org/martin/PorterStemmer

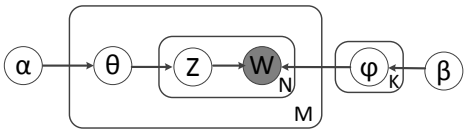


Fig. 1. Latent Dirichlet Allocation (LDA) plate model

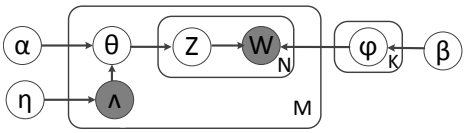


Fig. 2. Labeled Latent Dirichlet Allocation (LLDA) plate model

*2) Semantic-based Topics:* Many semantic tools have been developed based on top of large document corpus like Wikipedia, News, Blogs. Example of such tools include: Alchemy[3], Calais[4], Textwise[5], etc. They are built using statistical natural language processing and machine learning techniques. These tools are inherently capable of extracting the semantic concepts, identifying named entities, assigning an hierarchical category label, etc. to a document based on its content.

To create a user profile we first group all the tweets of a user $u_i$ into a single document $D_i$. We extract concepts and category labels from the document $D_i$ using language modeling and neural networks[6]. The semantic-based user profile ($U_i^{semantic}$) contains the keywords representing such concepts and category labels. While such user profile has least number of keywords as compared by other approaches, it remains easy to understand and interpreted by a human.

*3) Latent Dirichlet Allocation (LDA):* Latent Dirichlet Allocation (LDA) is an unsupervised learning algorithm that models each document in a corpus as a mixture of topics. The topics in turn are mixtures of words in the vocabulary. The latent variables of document to topics mixture distribution and topic to words mixture distribution are learned using the LDA technique.

Figure 1 shows the plate notation capturing the dependencies among different parameters of the model. $\alpha$ and $\beta$ are Dirichlet priors on per-document topic distributions and per-topic word distributions. $\theta_i$ represents the topic distribution for a document $D_i$, while $\phi_k$ represents the word distribution for topic-k. $w_{ij}$ and $z_{ij}$ represent the word and the topic of $i^{th}$ term in $j^{th}$ document. K represents the number of topics and M represents the number of documents in the corpus. Among many variables, only the words $w_{ij}$ are observed variables, while the remaining are latent/hidden variables. There are number of techniques for inferring the latent variables. In our current work we make use of collapsed Gibbs sampling [24] approach for inferring the latent variables of the corpus.

The output of the LDA learning process is topic-to-word distributions ($\phi_k$) and document-to-topic distributions ($\theta_i$). As for the frequency-based profiles, we extract top-K keywords (with $K$ ranging from 10 to 150) after combining both these distributions $\theta_i$ and $\phi_k$ for a given user $U_i$, and group these keywords and term them as LDA-based user profile ($U_i^{LDA}$).

*4) Labeled Latent Dirichlet Allocation (LLDA):* LDA is an extremely popular model for summarizing a document corpus. However, it is not designed to handle multiple-labeled corpora, and it also suffers from the fact that inferred topics are not labeled thus needing a human to create topic interpretations. Labeled LDA ($LLDA$) [21] is a generative model for document collections that have labels assigned to each of the document. Topics extracted using LLDA are inherently labeled using the labels supplied with the documents. The topic-word distributions inferred during the learning process correspond to the label topics. Each label will have a multinomial distribution over the words found in the corpus.

TABLE I. TOPIC KEYWORDS EXTRACTED FOR A POPULAR TWITTER USER AND STACKOVERFLOW CO-FOUNDER: JOEL SPOLSKY (@SPOLSKY)

| | |
|---|---|
| $U_i^{tf}$ | stack, overflow, google, app, software, developer, feature, generator, twitter, design, #stackoverflow, … |
| $U_i^{tfidf}$ | #annoyingsecurityquestions, #shootingfishinabarrel, #spoton, cinnabon, justintv,torah, #changetheratio, … |
| $U_i^{semantic}$ | stack exchange, computers, internet, protocols, arts, science fiction and fantasy, software, crafts, knitting and crochet, computers, open source, software |
| $U_i^{lda}$ | startup, social, facebook, business, obama, romney, google, … |
| $U_i^{llda}$ | development, sharepoint, serial, compression, ms, graph, graphics, uml, azure, scriptaculous, … |

Figure 2 shows the LLDA plate diagram. Most of the parameters are same as LDA parameters. Additionally we see variables($\eta$ and $\Lambda$) corresponding to the labels of the documents. In LLDA, the document is supervised to learn the topics corresponding to the attached labels. We use collapsed Gibbs sampling [24] for inferring the latent variables. Similar to LDA, we extract top-K keywords (with $K$ ranging from 10 to 150) after combining the document-label distribution ($\theta_i$) and label-word distribution ($\phi_k$), and group them as LLDA-based user profile ($U_i^{LLDA}$).

### B. Tweet Messages Classification

In this section, we address the problem of classification of a tweet message $T_j$ that contains an ambiguous company name and posted by an user $u_i$, on whether it is related to a company entity $C$. As discussed in Section II, we model the company entity $C$ as a weighted set of keywords, where keywords act as positive or negative evidence depending on their weights. The tweet bag of words are compared against the company entity $C$ bag of words. Depending on the amount of positive or negative keywords that are present in the tweet, it is classified as related to or not related to the company entity.

A tweet being a short message (maximum of 140 chars) would contain on average 10-15 words. As the tweet message contains very little context, the burden of better classification shifts to obtaining a better company entity $C$ description. We construct an entity profile $C$ following the findings of Yerva et al. in [28], where the authors identify multiple information sources to richly model the company entity profiles. They extract relevant keywords from the homepage[7] of the entity,

[3]http://www.alchemyapi.com/

[4]http://www.opencalais.com/

[5]http://www.textwise.com/

[6]See http://www.textwise.com/api/documentation/introduction

[7]Ex: http://www.apple.com for *Apple* company entity

keywords from the meta-data provided on the company web-pages, keywords from the glossary related to the category[8] of the company, keywords inferred using Google-set, or Wordnet services. They also rely on Wikipedia disambiguation pages for negative evidence keywords.

Moreover, the company entity profile $C$ should not have too few words, resulting in less overlap with the tweet message keywords, therefore leading to random classification of tweets. On the contrary, the entity profile should not be too general, therefore avoiding many false positives during classification.

For our classification problem, we make use of Naive Bayes Classifier [10], [13]. We assume the words appearing in a tweet independently contribute towards the evidence of whether the tweet belongs to the company, or not.

For each tweet $T_i = set\{wrd_j^i\}$ we compute the conditional probabilities $P(C \mid T_i)$ and $P(\overline{C} \mid T_i)$ for deciding if a tweet belongs to a company $C$ or not. We make use of Bayes theorem for computing these terms.

$$\begin{aligned} P(C \mid T_i) &= \frac{P(C) * P(T_i \mid C)}{P(T_i)} \\ &= \frac{P(C) * P(wrd_1^i, \ldots, wrd_n^i \mid C)}{P(T_i)} \\ &= K_1 \prod_{j=1}^{n} P(wrd_j^i \mid C) \end{aligned} \tag{4}$$

Similarly we have,

$$P(\overline{C} \mid T_i) = K_2 \prod_{j=1}^{n} P(wrd_j^i \mid \overline{C}) \tag{5}$$

where, $P(wrd_j \mid C)$ and $P(wrd_j \mid \overline{C})$ are the weights associated with the words $wrd_j$ as described in the previous section. Depending on whether $P(C \mid T_i)$ is greater than $P(\overline{C} \mid T_i)$ or not, the Naive Bayes Classifier decides whether the tweet $T_i$ is related to the given company or not, respectively.

Another way of improving the tweet message classification is through enriching the context of the tweet. While there is no clear concise definition of context, the location and the time of the tweet message, the previous and next messages (neighborhood) of the current message, etc. could act as context of the message. In this work we use the user profile constructed using the different techniques to provide certain context to the message to be classified.

A user profile $U_i$ corresponding to a user $u_i$, is modeled as a set of weighted keywords. We have already shown various techniques to construct such user profiles for the user generated content. When we combine the user context $U_i$ with the tweet message $T_j$ we get a new message, i.e., the tweet message in user context and we call it $M_j$. Even though there are many ways of combining the user profile $U_i$ and tweet message $T_j$ for obtaining $M_j$, we choose to focus on a simple union function. The resulting $M_j$ will contain all the keywords found in $U_i$ and $T_j$.

$$M_j = \cup\{T_j, U_i\} = Set\{\overbrace{w_1^j, \ldots, w_n^j}^{\text{Tweet words}}, \underbrace{w_1^i, \ldots, w_m^i}_{\text{User Profile Keywords}}\} \tag{6}$$

We again use Naive Bayes Classifier for classifying the context enhanced Twitter messages $M_j$. The conditional probabilities $P(C \mid M_j)$ and $P(\overline{C} \mid M_j)$, similar to eqns 4 and 5, decide if the original tweet $T_j$ belongs to the company entity $C$ or not.

$$P(C \mid M_i) = K_1 \overbrace{\prod_{k=1}^{n} P(w_k^j \mid C)}^{\text{Tweet Component}} \prod_{k=1}^{m} P(w_k^i \mid C) \tag{7}$$

$$P(\overline{C} \mid M_i) = K_2 \prod_{k=1}^{n} P(w_k^j \mid \overline{C}) \overbrace{\prod_{k=1}^{m} P(w_k^i \mid \overline{C})}^{\text{User Profile Component}} \tag{8}$$

### C. Cross Social Network User Profiles

More than just the features described above and their combination, we can exploit the fact that users participate on different social networks. Thus, we generate a *global social profile* that combines evidences from different social networks the user is involved in. This allows to take into account the diversity of content produced by users over different type of social networks (e.g., professional and leisure). By accounting the variety of content and meaning an entity can have for the user we aim at improving effectiveness of tweet classification. In the context of this paper, we combine a general-interest social network (Twitter) with a domain specific one (Stack-Overflow) to build more diverse user profiles. Such enhanced profiles, obtained by merging the keyword lists from the best performing technique on each network, prove to be very useful when the company profile $C$ is not extensive or noisy.

## IV. Experimental Evaluation

### A. Data Description

We applied the user profile techniques explained in Section III-A on both a Twitter and a Stack Overflow[9](SO) dataset. Stack Overflow is a website that features questions and answers on a wide range of topics in computer programming. Questions are tagged by the users (up to 5 tags)–at the moment of writing this paper, the top-6 tags on the website are: C#, Java, PHP, JavaScript, JQuery and Android. Stack Overflow embeds also a simple but very effective reputation system that contributed to the spam-free user experience on the website. For instance, questions can be re-tagged only by users with a reputation score above 500 (i.e., users who have spent a fair amount of time contributing to the platform). For this reason, we consider StackOverflow tags as a set of "labels" carefully redacted by domain experts, hence a valid input to our LLDA user profiling technique; e.g., once a user writes a valid answer to a question tagged as "Scala", we can indeed infer that she has some

[8]Apple is a Computer Technology category company.

[9]Stack Overflow: http://stackoverflow.com/

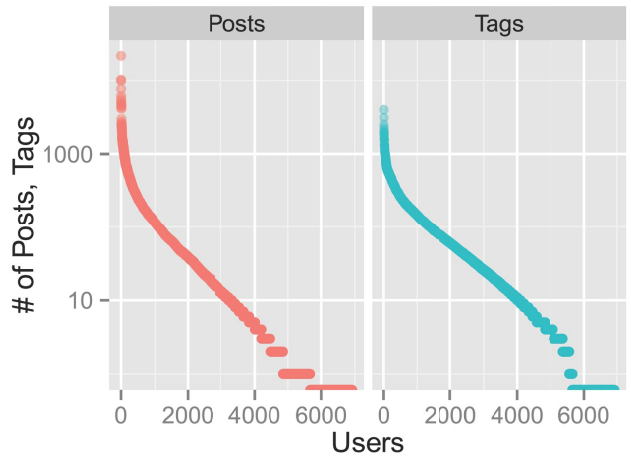


Fig. 3. Power-law distribution of Stack Overflow Posts and Tags

| | |
|---|---|
| Users | 6923 |
| StackOverflow posts | 592,021 |
| Distinct SO tags | 22,930 |
| Tweets | 4,894,944 |

TABLE II. STACKOVERFLOW + TWITTER DATASET STATISTICS

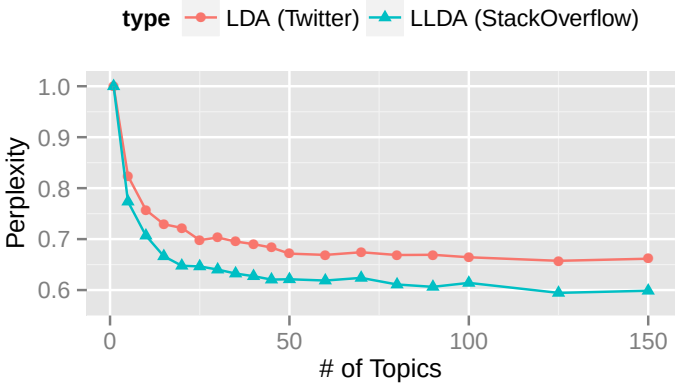


Fig. 4. Perplexity on the Twitter and StackOverflow corpora (normalized to 1)

expertise on the Scala programming language, hence defining a characteristic aspect of her profile.

Lacking a similar set of accurate labels for Twitter users, we employed LDA instead of LLDA. On the other hand, we applied TF, TF-IDF and Semantic on both datasets.

The evaluation dataset has been built with the following procedure:

- from the Stack Exchange Data Dump of August 2012[10], we identified 7772 users who reported their Twitter account in the Stack Overflow profile description
- for each of these users, we extracted all the data available in the StackOverflow XML dump: profile information, questions and answers, and tags (extracted both from the questions asked directly by the user and from the questions the user's answers referred to)
- we crawled Twitter (using the REST API) to obtain the latest tweets of the user (until Mar 12, 2013)

After cleaning the dataset (e.g., removing users with no activity, or with a protected Twitter account), we merged the information coming from both sources (Stack Overflow and Twitter) in a columnar database, to enable fast slicing and dicing of the user data.

It is worth to note that, due to the rate limiting in the Twitter REST API, we collected a maximum of 1000 tweets per user. On the other hand, the Stack Exchange Data Dump allowed us to process the whole history of the Q&A platform.

Our sample of the Stack Overflow users' activities follows a power-law, as shown in Figure 3. Such distribution is very common in websites driven by user-generated content, confirming the validity of the approach followed to build our dataset.

[10]http://www.clearbits.net/creators/146-stack-exchange-data-dump

### B. User Profiles Construction

For each user, we extracted the text content of her tweets and StackOverflow content, and used it as an input for the 5 techniques explained in Sec. III-A: TF, TF-IDF, LDA, LLDA, Semantic. While TF, TF-IDF and Semantic were applied on both social networks, we used LDA exclusively on Twitter, and LLDA exclusively on StackOverflow. TF, TF-IDF and Semantic return a ranked list of tokens, and for each we extracted the top-K results, with $K \in \{10, 25, 50, 75, 100, 125, 150\}$. LDA and LLDA, instead, required a more elaborated procedure. First, we computed the perplexity score for each model, varying the number of extracted topics. The perplexity score[11] measures how much the original corpus differs from one generated by the model trained on such corpus. Although it is expected that the perplexity score decreases with a higher number of topics, it does not give any guarantees on the quality and coherence of the topics. Furthermore, training a LDA or LLDA model does not scale gracefully with the number of topics (both in terms of CPU time and memory required). Given the results shown in Fig. 4, and after manual inspection of the generated topics, we opted to train our models with 50 topics, as it represented a good tradeoff between time spent by the training procedure and quality of the topics. Once the topics are generated, we run the inference process on the data of each single user, obtaining a ranked list of tokens which we sliced to extract the top-K keywords (with $K \in \{10, 25, 50, 75, 100, 125, 150\}$).

Fig. 5 reports the average overlap between profiles extracted for a single user on both Twitter and StackOverflow. The overlap has been computed in the following way: for each user, we extract 2 top-K lists from both social networks, employing TF, TF-IDF or (respectively) LDA and LLDA. We then compare the two lists with the following similarity

[11]http://en.wikipedia.org/wiki/Perplexity

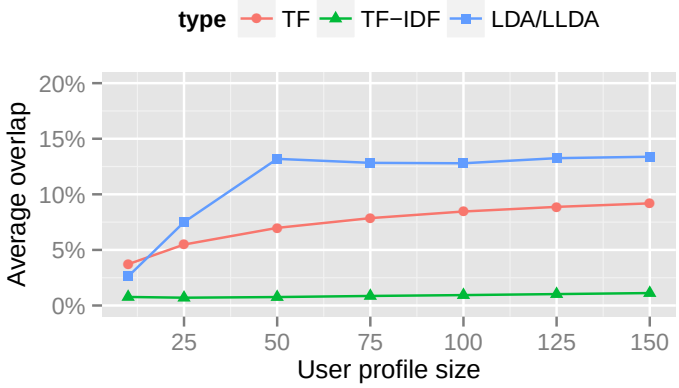


Fig. 5. Average overlap between User Profiles extracted from Twitter and StackOverflow.

function:

$$Similarity = \frac{\sum_{i=1}^{K} get\_close\_match(topK\_TW[i], topK\_SO)}{K} \quad (9)$$

`get_close_match` is a function that returns 1 when it finds a fuzzy match between one of the tokens in the Twitter top-K and the StackOverflow top-K, 0 otherwise. The fuzzy matching is mostly based on the concept of string edit distance (i.e., Levenshtein distance), but the cutoff parameter has been set in such a way that almost only perfect matches would return a 1.

"Semantic" is not included in the Fig. 5 because the technique we use does not return large sets of concepts, hence we cannot build Semantic profiles of different sizes. Similarly to LDA/LLDA though, the Semantic profiles are characterized by an average $11\%$ overlap between Twitter and StackOverflow.

The relatively small overlap of the profiles built on different social networks is very valuable in our scenario, because it improves the diversity of the keywords used to disambiguate the tweets, as explained in the following section. It is also remarkable that, no matter the bias of our dataset towards high-tech oriented users, the profiles built on Twitter and StackOverflow show very different facets of the user.

### C. Tweet Message Classification

WePS-3 dataset[12] contains tweets related to 100 company names, with an average of 500 tweets for each company name. The ground truth for each of this tweet is available in the dataset. However, we could not use WePS-3 for our experiments, because most of its tweets have not been posted by the users in our dataset. In fact, for comparison with our techniques, we need both the tweet message and the user who posted that message.

From the $\sim$5 Million tweets we collected, we coose a subset of those tweets that contained at least one of the following set of six words: *apple*, *oracle*, *apache*, *subway*, *seat*, *orange*. The WePS-3 dataset contains 100 company names, with varying degree of ambiguity. We chose 6 company names as a representative sample of the entire dataset. Each of these 6 company names have multiple interpretations; e.g., the *apple* keyword could mean a fruit, the Apple company, New York city, etc. We are interested in classifying the tweet containing one of this keyword (for example: *subway*) with respect to its reference (or not) to the actual company (e.g., the *Subway* fast-food franchise).

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

TABLE III. DATASETS COMPARISON: PERCENTAGE OF TWEETS, CONTAINING THE COMPANY KEYWORD, THAT ARE RELATED TO THE COMPANY ENTITY.

| Dataset | apple | oracle | apache | subway | orange | seat |
|---|---|---|---|---|---|---|
| WePS3 | 0.83 | 0.78 | 0.47 | 0.45 | 0.05 | 0.02 |
| SOTW | 0.93 | 0.96 | 0.97 | 0.12 | 0.15 | 0.01 |

For each of these 6 keywords, we manually annotated a total of 100 tweets, stating if they were related (or not) to their company entity. We refer to this dataset as the SOTW dataset. This manual annotation would act as ground truth for verifying the classification results of the two different approaches: one with the classifier that takes the user profile into consideration, one with the classifier that does not.

Table III shows the percentage of tweets that belong to the company entity in the two different datasets: WePS-3 dataset and our dataset(SOTW). It is interesting to observe that the related percentages for tech company names (*apple*, *oracle* and *apache*) are higher in our dataset when compared to WePS-3. This is due to the fact that SOTW contains mostly tech-savvy users, while WePS3 is formed by a more general audience. Therefore, knowing the context in which a tweet was posted reduces the ambiguity in its interpretation.

Next, we compare the performances of the two classifiers: (1) Base Classifier($BC$): the classifier which classifies tweets only based on the tweet keywords and the company entity keywords, (2) Enhanced Classifier($EC$): the classifier that considers user profile keywords along with the tweet and company entity keywords for its classification task.

The performance of the classifiers depends on: a) the quality and size($K$) of the user entity profile $U_i$; b) the size of the company entity profile $C$; and c) the percentage of tweets that contain overlapping words with the company profile words. We make use of the company entity profiles that were used in [28], [29]. As these company profiles were developed in the context of the WePS3 task [3], we assume that their accuracies are bounded by the values in the first row of Table III. Given the full-size company entity profile, we plot the accuracies of the classifiers by varying the number of words in the user profile, as shown in Figure 6. At $K = 50$, most of the user profiling techniques saturate the achievable accuracy of the classifier, suggesting that the user profile has already gathered a good candidate set of words for entity disambiguation. For this reason, we use $K = 50$ as the size of the user profile $U_i$, as it represents a good tradeoff between performance and computational cost.

We define the accuracy metric for the classifier as the percentage of tweets that are correctly classified. The performance of the classifier depends on the quality of the company entity profile $C$. Table IV shows the accuracies of the different classifiers, for a fixed size company profile and a size of $K = 50$ of the user profile. Given a fixed company entity

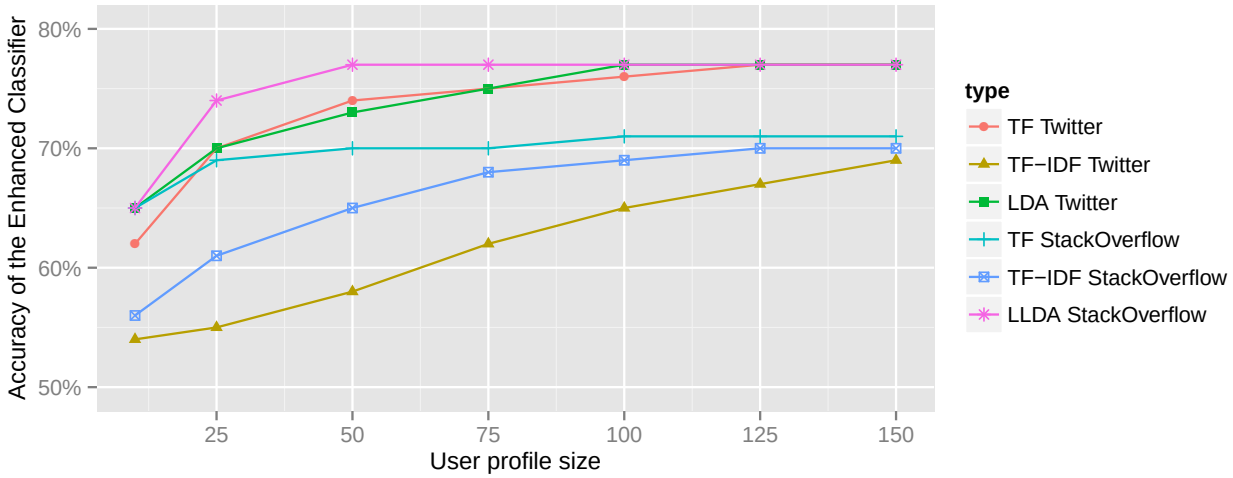


Fig. 6. Enhanced Classifier performance with different User Profile techniques and sizes.

TABLE IV. ACCURACY OF THE DIFFERENT CLASSIFIERS: BASIC CLASSIFIER AND ENHANCED CLASSIFIERS. STATISTICALLY SIGNIFICANT IMPROVEMENT OF EC OVER BC ARE INDICATED BY * (T-TEST $p < 0.05$)).

| | **BC** | **Enhanced Classifiers(EC)** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | **Twitter** | | | | **StackOverFlow** | | | | |
| **Company** | Basic | TF | TFIDF | Semantic | LDA | TF | TFIDF | Semantic | LLDA | Hybrid |
| apple | 0.55 | 0.83 | 0.58 | 0.77 | 0.83 | 0.76 | 0.71 | 0.69 | 0.83 | 0.83 |
| apache | 0.5 | 0.52 | 0.51 | 0.51 | 0.52 | 0.52 | 0.51 | 0.51 | 0.53 | 0.53 |
| oracle | 0.55 | 0.77 | 0.64 | 0.55 | 0.78 | 0.7 | 0.66 | 0.58 | 0.78 | 0.78 |
| orange | 0.5 | 0.54 | 0.51 | 0.51 | 0.54 | 0.53 | 0.53 | 0.51 | 0.55 | 0.55 |
| subway | 0.54 | 0.94 | 0.68 | 0.82 | 0.95 | 0.83 | 0.78 | 0.57 | 0.95 | 0.95 |
| seat | 0.52 | 0.81 | 0.56 | 0.59 | 0.76 | 0.84 | 0.71 | 0.55 | 0.96 | 0.98 |
| AVG | 0.53 | 0.74* | 0.58 | 0.63 | 0.73* | 0.70* | 0.65* | 0.57 | 0.77* | 0.77* |
| p-values | | 0.019 | 0.053 | 0.104 | 0.020 | 0.022 | 0.021 | 0.093 | 0.020 | 0.021 |

TABLE V. PERCENTAGE OF NON-OVERLAPPING TWEETS WITH THE COMPANY ENTITY PROFILE. THIS PERCENTAGE OF TWEETS WILL BE RANDOMLY DECIDED BY THE CLASSIFIERS. USER PROFILES CONTAIN K=50 KEYWORDS.

| | **BC** | **Enhanced Classifiers(EC)** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | **Twitter** | | | | **StackOverFlow** | | | | |
| **Company** | Basic | TF | TFIDF | Semantic | LDA | TF | TFIDF | Semantic | LLDA | Hybrid |
| apple | 84 | 0 | 75 | 16 | 0 | 20 | 35 | 40 | 0 | 0 |
| apache | 83 | 2 | 59 | 49 | 1 | 24 | 50 | 62 | 0 | 0 |
| oracle | 82 | 1 | 49 | 82 | 0 | 28 | 41 | 70 | 0 | 0 |
| orange | 87 | 13 | 76 | 66 | 11 | 21 | 33 | 75 | 0 | 0 |
| subway | 90 | 1 | 58 | 28 | 0 | 26 | 36 | 83 | 0 | 0 |
| seat | 95 | 34 | 86 | 80 | 45 | 29 | 56 | 89 | 4 | 0 |

profile $C$, we see that the enhanced classifiers($EC$) (that take user context into consideration) are outperforming the basic classifier($BC$). The results in Table V and IV clearly show that the user context helps the classifier in resolving the ambiguity involved in the company name. The percentage of tweets that do not overlap with the company profile in the test set represent the main cause of erroneous classifications.

In Table V, we show the percentage of tweets in the dataset that do not have any overlapping keywords with the company profile $C$. The higher the number, the lower is the chance for a classifier to make accurate classifications. We see that the column-1 (basic) has the highest number of such tweets, while the remaining columns (that represent the tweets enhanced with user context) have a very low non-overlapping number of tweets. The Enhanced Classifiers are in a better position to classify the tweets more accurately, thus achieving our goal of "making sense of the microposts".

Finally, we control the quality of the company profile $C$ by varying its size, whose impact on classifier performance is shown in Table VI, along with the p-values (two tailed t-test). We observe that each of the Enhanced Classifiers ($EC$) is performing better than the Basic Classifier($BC$), and this is true for all the size variations of the company profile. However, the percentage of improvement is statistically significant for lower sizes of the company profile. As it is relatively difficult to have an accurate company profile, based on our results we can benefit of the user social profiles especially when the company profile is noisy or too small.

Tables IV, V and VI report also the results for a *Hybrid* technique, which merges the best techniques from multiple social networks to obtain a more diverse user profile. On our dataset, we observe that term frequency (TF) is best among the techniques applied on Twitter, and LLDA based is best among the techniques applied on StackOverflow. The resulting Hybrid user profile is then the top-25 for Twitter TF, combined with the top-25 LLDA for StackOverflow. Although the improvement of the Hybrid classifiers is not statistically significant on

TABLE VI. AVERAGE ACCURACY MEASURE, ALONG WITH P-VALUES, FOR THE DIFFERENT CLASSIFIERS W.R.T. VARYING QUALITY OF THE COMPANY PROFILES. STATISTICALLY SIGNIFICANT IMPROVEMENT OF EC OVER BC ARE INDICATED BY * (T-TEST $p < 0.05$)).

| Number of words in a Company Profile | BC | Enhanced Classifiers(EC) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Twitter | | | | StackOverFlow | | | | |
| | Basic | TF | TFIDF | Semantic | LDA | TF | TFIDF | Semantic | LLDA | Hybrid |
| 0 | 0.500 | 0.500 | 0.500 | 0.500 | 0.500 | 0.500 | 0.500 | 0.500 | 0.500 | 0.500 |
| 100 | 0.527 | 0.735* (0.019) | 0.580 (0.053) | 0.625 (0.104) | 0.730* (0.020) | 0.697* (0.022) | 0.650* (0.021) | 0.568 (0.093) | 0.767* (0.020) | 0.770* (0.021) |
| 200 | 0.562 | 0.770* (0.020) | 0.630 (0.071) | 0.675 (0.079) | 0.768* (0.020) | 0.710* (0.019) | 0.692* (0.021) | 0.608 (0.079) | 0.768* (0.020) | 0.770* (0.020) |
| 500 | 0.607 | 0.770* (0.039) | 0.677 (0.084) | 0.708 (0.078) | 0.768* (0.038) | 0.727* (0.037) | 0.713* (0.035) | 0.683 (0.078) | 0.768* (0.038) | 0.770* (0.039) |
| 1000 | 0.633 | 0.770 (0.064) | 0.702 (0.130) | 0.713 (0.100) | 0.768 (0.063) | 0.730 (0.068) | 0.718 (0.058) | 0.693 (0.113) | 0.768 (0.063) | 0.770 (0.064) |

SOTW, we argue that its main advantage is represented by its reliable performance, regardless of the quality of the company profile. Our speculation is that, on a larger and more diverse dataset, the Hybrid classifier would systematically outperform the other Enhanced classifiers.

## V. RELATED WORK

*a) Topic Modeling in Micro-blogging Platforms:* A number of recent works have explored the use of topic models in the Twitter domain for modeling Twitter messages and users [11], finding topical authorities [19], [25], making recommendations [8], and comparing it with other media [7], [30]. We also focus our attention on works that have explored user modeling [1], [7], [9], [2] in micro-blogging platforms.

Works like, for example [14], [22], have focused on adapting techniques and tools that were successful on text corpora to the recent vastly popular micro-blogging platforms. They adapted the named entity extraction (NER) techniques for the shorter and noisy micro-blog posts. The NER task is a critical step for the the task of identifying the subset of tweets that are relevant to an entity which we tackle in our paper.

Topic modeling of Twitter messages has been considered in [11], where models for three different tweet aggregation strategies have been considered: First, each Twitter message is considered as a document; second, all the tweets corresponding to a user are considered as being a single document; and finally, all tweets containing a particular term are put together in a one single document. These three strategies are referred to as MSG-Topic-Model, USR-Topic-Model and TERM-topic model. They show that the topics learned by the various schemes are different in quality. The topic models learned from aggregated messages of a user can lead to superior performance in classification problems. In our current work we grouped all the tweets corresponding to a user in to a single document and used it to infer the users' topics.

Several previous works [19], [25], [20] have used topical modeling features on micro-blogging platforms for finding topic-based experts and authorities. The authors in their work on topical authorities in microblogs [19] propose various sets of features in order to find topic-based authoritative users. The set of features are based on how frequently users tweet, what percentage of their tweets are retweets, how often their tweets are retweeted, how often users are mentioned by other users, and how diverse or focused are the tweets to a particular topic. TwitterRank [25] proposes a ranking algorithm, an adaptation of PageRank algorithm, for finding topic-sensitive influential users. They make use of LDA on the twitter content for linking an user with certain set of topics, and use topic level similarity among users as feature of their ranking algorithm.

Most user interactions in Twitter are still primarily focused on the social graphs. Characterizing micro-blogs with topic models [20] explores content analysis of Twitter feeds for addressing special information needs of the users. They apply LDA [4] and labeled LDA [21] for identifying the latent topics of Twitter messages. Using unsupervised LDA they assign latent topics into one of the four subcategories {*substance*, *social*, *status*, and *style*}. The partially supervised labeled LDA could assign labels (emoticons, hashtags, etc.) to the latent topics extracted from the Twitter feeds. We apply similar techniques for the problem of tweet disambiguation.

Some works, as in [30], [7], have relied on topic modeling for comparing recent micro-blogging platforms and traditional news media platforms. In the paper [30], the authors do an empirical comparison of the Twitter content with that published on tradition media like the New York Times. Using standard LDA they infer topics from the news dataset, while they propose a Twitter-LDA model for extracting topics from Twitter data. This study shows how certain topics are popular on Twitter while some others are popular on news media. In [7] the authors extend their user modeling framework [1] for comparing the usage behavior on two popular micro-blogging platforms: Sina Weibo[13] and Twitter.

*b) User Modeling over Micro-blogging platforms:* Web is gradually transforming itself as a users personal archive, where users not only find information but leave, share and archive information [15]. Twitter being widely adopted, real time and representative of the users, despite being of noisy nature, is a great source for modeling a user [27]. User profiles were constructed in [23], [1], [9] for better news and people-to-follow recommendations, dealing with information overload, understanding users' expertise and interests, etc. [23] make use of entity profiles, that are sets of information extracted for each ambiguous person in the entire document, and features based on topic models to cluster documents –containing a person name– based on the actual person entity. Authors of [1] analyze user modeling on Twitter for personalized news recommendations. Their framework helps in creating user profiles that are based on extracted topics and entities from the tweet content, and show its superior performance compared to hash-tag based user profiles. They also consider temporal aspects of the user profile for better news recommendations.

The authors of [9] propose techniques to construct multi-faceted user profiles for Twitter users, thereby helping one to navigate the complex domain-space represented by Twitter. Their model profiles users and their social networks using tags and labels from curated lists. The work [2] extracts professional interests from social web(Facebook, Twitter) profiles.

[13]http://www.weibo.com

Twittomender [8] explores building of user profiles based on tweets which are grouped as users' own tweets, followers tweets and followees tweets. They make use of TF-IDF ranking technique in construction of the user profile, which they use for recommending other Twitter users to follow.

*c) Micro-post Classification:* In [12], [26] the authors present LDA transfer learning. Transfer Learning is the process of generic learning in one domain and applying the model in a different domain. In topic-bridged LDA ($tLDA$) a model is built from a variety of labeled and unlabeled documents, and they apply transfer learning for document classification task. One of our technique ($LLDA$) is based on transfer learning.

Several works[3], [28], [29] have addressed the problem of tweet classification in various contexts. For example, [28], [29] addresses the problem of Entity-based classification of tweets. Their techniques focus on accurately building the company entity profile, they also rely on *relatedness factor* metric of the company, and adapt active-learning for continuously improving their company entity profile. In our work, we focus on improving the classifiers performance by enriching the context of the tweet messages using the user social profiles.

## VI. Conclusions

Users in on-line social network generate content based on their interests and knowledge. They refer to entities which, in the given context are unambiguous for the other users who are consuming the content. However, to enable applications such as entity-centric search over social network content, we need to disambiguate the user generated content. In this work we presented a number of techniques for constructing user entity profiles, and evaluated their effectiveness for the tweet disambiguation task. Such user entity profiles present a summarized view of the user generated content across various social networks. In the second part of the paper we have shown the importance of context in handling the tweet ambiguity: We used the user entity profiles to provide the missing context to the microposts, thus seeing an improved performance of the tweet classifier. Specifically, frequency-based features on Twitter and LLDA features on StackOverflow give user profiles that significantly improve effectiveness of disambiguation as compared to baseline approaches. Moreover, we have observed that the most reliable results are obtained by the combination of such best performing techniques to generate a global user profile that combines evidences from different social networks the user is involved in. In the current work we focused only on the user generated content, however, in future work we want to consider other information like the users social connections and their activities on the social networks for constructing better user profiles.

## Acknowledgments

This work was supported in part by OpenIoT (EU-FP7 project) and MEM0R1ES (Hasler Foundation project).

## References

[1] Fabian Abel, Qi Gao, Geert-Jan Houben, and Ke Tao. Analyzing user modeling on twitter for personalized news recommendations. UMAP, 2011.

[2] Fabian Abel, Eelco Herder, and Daniel Krause. Extraction of professional interests from social web profiles. In *UMAP*, 2011.

[3] E. Amigo, J. Artiles, J. Gonzalo, D. Spina, B. Liu, and A. Corujo. WePS3 Evaluation Campaign: Overview of the On-line Reputation Management Task. In *WePS*, 2010.

[4] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 2003.

[5] David G. Brizan and Abdullah U. Tansel. A Survey of Entity Resolution and Record Linkage Methodologies. *IIMA*, 2006.

[6] Chia-Hui Chang, Mohammed Kayed, Moheb Ramzy Girgis, and Khaled Shaalan. A survey of web information extraction systems, 2006.

[7] Qi Gao, Fabian Abel, Geert-Jan Houben, and Yong Yu. A comparative study of users' microblogging behavior on sina weibo and twitter. In *UMAP*, 2012.

[8] John Hannon, Mike Bennett, and Barry Smyth. Recommending twitter users to follow using content and collaborative filtering approaches. In *RecSys*, 2010.

[9] John Hannon, Kevin McCarthy, Michael P. O'Mahony, and Barry Smyth. A multi-faceted user model for twitter. In *UMAP*, 2012.

[10] David Heckerman. *Learning in graphical models*, chapter A tutorial on learning with Bayesian networks. MIT Press, 1999.

[11] Liangjie Hong and Brian D. Davison. Empirical study of topic modeling in twitter. In *SOMA*, 2010.

[12] Jeon-Hyung Kang, Jun Ma, and Yan Liu. Transfer topic modeling with ease and scalability. *CoRR*, 2013.

[13] David D. Lewis. Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval. In *ECML*, 1998.

[14] Chenliang Li, Jianshu Weng, Qi He, Yuxia Yao, Anwitaman Datta, Aixin Sun, and Bu-Sung Lee. Twiner: named entity recognition in targeted twitter stream. In *SIGIR*, 2012.

[15] Sian Lindley, Cathy Marshall, Richard Banks, Abigail Sellen, and Tim Regan. Rethinking the web as a personal archive. In *WWW*, 2013.

[16] Bing Liu and Lei Zhang. A survey of opinion mining and sentiment analysis. *Mining Text Data*, 2012.

[17] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Linguisticae Investigationes*, 2007.

[18] Niels Ott, Ramon Ziai, Michael Hahn, and Detmar Meurers. Comet: Integrating different levels of linguistic modeling for meaning assessment. In *(SemEval)*, 2013.

[19] Aditya Pal and Scott Counts. Identifying topical authorities in microblogs. In *WSDM*, 2011.

[20] Daniel Ramage, Susan Dumais, and Dan Liebling. Characterizing microblogs with topic models. In *AAAI*, 2010.

[21] Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D. Manning. Labeled LDA: a supervised topic model for credit attribution in multi-labeled corpora. EMNLP, 2009.

[22] Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. Named entity recognition in tweets: an experimental study. In *EMNLP*, 2011.

[23] Harish Srinivasan, John Chen, and Rohini Srihari. Cross document person name disambiguation using entity profiles, 2009.

[24] Stanford topic modeling toolbox. http://nlp.stanford.edu/software/tmt/tmt-0.4/.

[25] Jianshu Weng, Ee-Peng Lim, Jing Jiang, and Qi He. Twitterrank: finding topic-sensitive influential twitterers. In *WSDM*, 2010.

[26] Meng-Sung Wu and Jen-Tzung Chien. A new topic-bridged model for transfer learning. In *ICASSP*, 2010.

[27] L. Yang, M. Moshtaghi, B. Han, S. Karunasekera, R. Kotagiri, T. Baldwin, and A. Harwood. Mining micro-blogs: Opportunities and challenges. Social Networks: Computational Aspects and Mining.

[28] Surender R. Yerva, Zoltan Miklos, and Karl Aberer. What have fruits to do with technology? the case of Orange, Blackberry and Apple. In *WIMS*, 2011.

[29] Surender R. Yerva, Zoltan Miklos, and Karl Aberer. Entity-based classification of twitter messages. *International Journal of Computer Science and Applications*, 2012.

[30] Wayne X. Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee P. Lim, Hongfei Yan, and Xiaoming Li. Comparing twitter and traditional media using topic models. In *ECIR*, 2011.