Detecting Topic Labels for Tweets by Matching Features from Pseudo-Relevance Feedback

Jing Zhang **Derek Liu** **Kok-Leong Ong** **Zhijie Li** **Ming Li**

School of Information Technology, Deakin University
221 Burwood Highyway, Burwood, Victoria 3125
Email: {jing.zhang, derek.liu, kok-leong.ong, z.li, ming.li}@deakin.edu.au

Abstract

Detecting a suitable topic label for short texts, e.g., tweets from Twitter, is an important component in many applications including diversity ranking, clustering, information retrieval, and information filtering. To automatically detect topic labels however is a major challenge. The character limit of a short text means the lack of a significant feature space to adequately describe its content in relation to other short texts in a given collection. Therefore, methods like LDA, TF-IDF or similarity measures all fail due to their sensitivity to a small feature space. And when a collection of related short texts are considered, e.g., from a Twitter search, the result set collectively exhibits sparsity *and* high dimensionality – a nightmare for information processing. A solution to this problem is to expand the feature space through a process known as pseudo-relevance feedback. Unfortunately, they disappoint when subjected to real-world conditions. The fundamental problem lie in the level of noise present in both the short texts and the feedback source, which is often the World Wide Web. We propose a novel pseudo-relevance feedback algorithm to accurately identify topic labels for short texts. Our algorithm robustly handles noise in both the short texts and the feedback source through a method called 'feature matching'. Empirical results confirm the efficacy of our algorithm.

Keywords: Tweets, Twitter, Pseudo-Relevance Feedback, Short Texts, Topic Detection

1 Introduction

The modern Web is no longer just a repository for Web documents. It is now a hybrid of different media and different Web applications. Most recently, a huge amount of user generated content arising from social networking Websites are fuelling a new category of data. They are large in volume but each is terse in its content. We call them short texts. Short texts are increasingly becoming prevalent on the Web. They exist as summaries to a Website in search results, as tweets on Twitter, as status updates on FaceBook, or as comments on YouTube.

The volume of short texts has motivated many applications requiring the use of algorithms in areas such as diversity ranking, clustering, classification, information retrieval, and information filtering. These algorithms in turn depend on core components, one of which is to know the topic label of a short text. For example, some diversity ranking algorithms achieve diversity by ensuring different topics of short texts are included. Another example would be in classification, where a rank of topic labels is used to classify short texts into pre-determined categories.

Topic detection in short texts however is a challenging problem. Using the case of tweets for example, the 140 character limit means that there is hardly sufficient features present to adequately describe its content in relation to other tweets in a given collection. When the feature space is very small and the collection in question creates a collective feature space that is very sparse and high in dimension, most techniques like LDA (Blei , Ng & Jordan 2003), TF-IDF (Manning , Raghavan & Schtze 2008), or feature-based similarity measures would all fail under real-world conditions. This has been well-reported in many other literature such as (Bernstein *et. al.* 2010) and (Zhang *et. al.* 2011).

A way to overcome the limitation of small feature spaces and to deal with a collection that is sparse and highly dimensioned is to expand (or enrich) the original feature space by adding related features from another source. This technique is known as pseudo-relevance feedback, or simply relevance feedback (Lloret 2009). The feedback source, which is where additional related features are found, can be

- a collection of other short texts that has been manually processed;
- a collection of well-structured documents in the same domain as the short texts;
- a public domain collection such as Wikipedia or WordNet;
- or the largest public domain resource, i.e., World Wide Web.

If we consider short texts such as those drawn from Twitter, then the first two feedback sources will not be practically feasible because (i) of the effort required to build the short texts collection or the well-structured documents; and also (ii) the feedback source is likely to become outdated quickly when we consider how fast tweet topics may change. The third feedback source, although more robust towards changes, can be limited in the scope of topics it can cover. The last feedback source, the World Wide Web, is the largest public domain resource and is likely to evolve as rapidly as the topics developing on Twitter. So theoretically, the Web is the ideal candidate.

In exploring our solution, we came across feedback systems that uses the Web as its feedback source. The most recent is the work reported in (Bernstein *et. al.* 2010), which is also very close to the problem we are trying to solve. We recreated this system based on the description given and discovered that when the Web is used as the feedback source, the results can disappoint when real-world tweets are used. The problem lie in the noise level of the feedback source, which we will discuss in detail next. Nevertheless, the poor results motivated us to search for a solution that would perform well in real-world situations.

Our quest, based on an understanding of the issues surrounding the Web as a feedback resource, saw the development of a *feature matching* algorithm that would produce an accurate way to determine the topic label of a tweet. The prototype of our implementation is now live for public testing and the evaluation of user results has confirmed its ability to deliver a high level of accuracy based on users of Twitter.

We shall now introduce our *feature matching* algorithm in Section 3 but before that, we discuss in Section 2 why current Web-based feedback systems fail to produce adequate results. We then present our experimental results in Section 4, where we compare the topic labels detected from our algorithm against other Web-based feedback systems. We then end this paper by pointing readers to related works in Section 5 and drawing our conclusions in Section 6.

2 Relevance Feedback in the Real-World

To understand why the state of the art in Web-based pseudo-relevance feedback fail, we discuss an implementation call Eddi (Bernstein *et. al.* 2010). Eddi was designed as a tool to organise tweets by their topics. To do so, a relevance feedback process was used to compute a topic label for the tweet. Tweets with similar topic labels are then grouped together.

Eddi's algorithm consists of three main steps: (i) text transformation, (ii) search engine query, and (iii) text feature extraction. The first step aims to transform a tweet into a search query. This involves basic pre-processing such as removing 'RT' (re-tweets), '@username' mentions, URL references, etc. The second step involves taking the transformed tweet and converting it into a search query. In (Bernstein *et. al.* 2010), this is done by identifying the noun phrases as it was found that nouns are good topic markers in long documents (Bendersky and Croft 2008, Hulth 2003). To find the nouns, a Part-of-Speech (POS) tagging software is used (Kristina and Christopher 2000). The nouns identified are then used to query a search engine - in their case, Yahoo!. The top ten Web documents associated with the nouns are then retrieved. Each Web document is then computed for its TF-IDF (i.e., *term frequency-inverse document frequency*) and the top TF-IDF (Manning , Raghavan & Schtze 2008) terms are then merged through a voting system, where terms more common among the ten documents are selected over terms with fewer votes. These terms are the topic label(s) associated with the tweet.

Let's look at a tweet that was handled well by Eddi: *"awesome article on some SIGGRAPH user interface work: http://bit.ly/30MJy"*. As per the algorithmic steps, the transformed output presents us with the search phrase (consisting of noun terms): "article SIGGRAPH user interface work". The first ten Web documents obtained from the search phrase are then downloaded and the TF-IDF of each term across the documents computed. The top TF-IDF terms obtained in this specific case were *animation*, *character*, *3D*, *computer*, *graphics*, *user*, *interface* and *SIGGRAPH*. These terms were clearly good candidates as topic labels for the original short text (tweet).

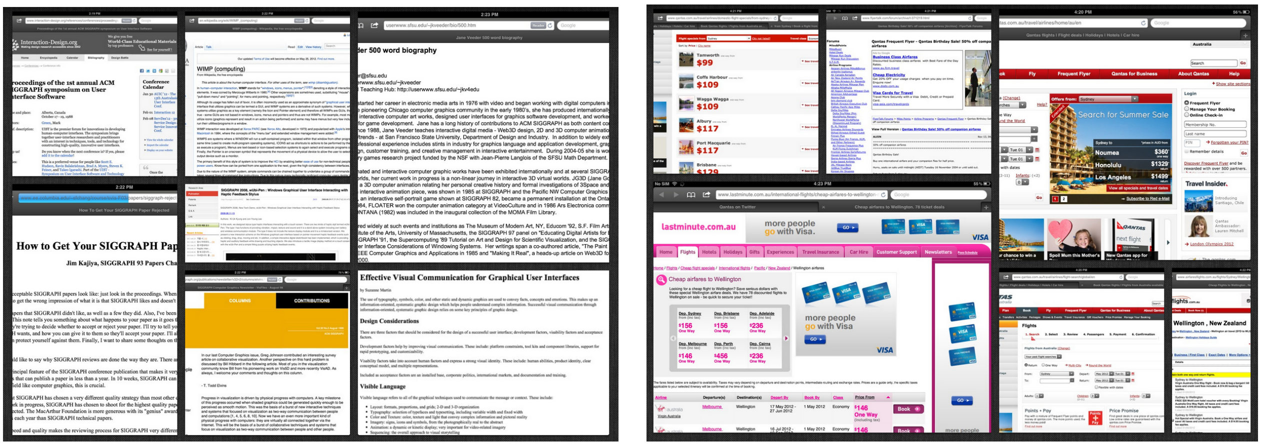
To see why this specific case works, we look at the results from the search engine (our feedback source) as shown in Figure 2. For the SIGGRAPH example, i.e., Figure 2(a), the documents returned are close to plain text which makes them easy to process. Compared to the Web documents we obtained from the next example shown in Figure 2(b), there is a sharp contrast in the level of 'noise' between the two sets of Web documents. For the SIGGRAPH tweet, the query returns the following URLs.

- `http://www.interaction-design.org/references/conferences/proceedings_of_the_1st_annual_acm_siggraph_symposium_on_user_interface_software.html`
- `http://www.interaction-design.org/references/conferences/proceedings_of_the_3rd_annual_acm_siggraph_symposium_on_user_interface_software_and_technology.html`
- `http://en.wikipedia.org/wiki/WIMP_(computing)`
- `http://www.siggraph.org/publications/newsletter/v32n3/columns/elvins.html`
- `http://kyungku.net/xe/publication/6442`
- `http://www.ee.columbia.edu/~sfchang/course/svia-F03/papers/siggraph-reject-how.htm`
- `http://mi-lab.org/about/people/michael-haller/`
- `http://web.cs.wpi.edu/~matt/courses/cs563/talks/smartin/int_design.html`
- `http://plecebo.org/content/fun-ui-innovations-siggraph-09-conference`
- `http://userwww.sfsu.edu/~jkveeder/bio/500.htm`

Now compare this to a tweet about Qantas, Figure 2(b): "*Sale #airfare #fly #Canberra to #Wellington from $410 with Qantas - http://t.co/2jsXBRbv*" which after the POS tagging, we had the search phrase "sale airfare canberra wellington qantas". The ten Web documents we obtained for this case contain JavaScripts, Flash content, advertisements, CSS styling, animated menus, dynamic presentation structures, dynamic forms, and server-side generated content. With so many layers of 'noise', any attempt to get to the actual content relevant to the search query becomes very challenging. We also went further by developing variations of Eddi such as (i) taking advantage of any short URLs present in the tweet to compute the TF-IDF; (ii) using a constrained set of Web documents (BlogSpot) to limit the level of noise; and (iii) using algorithms such as NReadability to extract the content. Unfortunately, the results we obtained from our experiments on all the variations were unsatisfactory. We conclude that when presented with such noisy documents, Eddi fails to provide accurate results. And with most of the Web documents today looking more like those seen in our Qantas example, the ability for Eddi to extend to real-world usage is actually questioned.

3 Feature Matching as Proxy Measure

Having failed from attempts to improve Eddi through various 'de-noising' strategies, we conclude that we have to accept the presence of noise in a feedback source like the Web. We also conclude that it would be difficult to overcome noise. This led us to a different strategy, where we embrace the noise present in Web documents instead. The idea in Eddi is to compute the TF-IDF from Web documents so as to



(a) (b)

Figure 1: (a) A montage of the various screens for the search phrase "article SIGGRAPH user interface work". Notice that the Web documents for this particular instance is not "noisy" and many of them are simple text-oriented documents without formatting, layers, advertisements, etc. Consequently, this makes extraction of the actual body of content easy and lowers the error probability significantly to allow the TF-IDF compute to show meaningful topic labels. (b) A montage of the various screens for the search phrase "sale airfare canberra wellington qantas". Compared to (a), the Web documents here are a lot more complex in their presentation as they incorporate dynamic content such as Flash and JavaScript, CSS styling and interactive menu, advertisements, photos and forms, etc. Extracting the main content from these Web documents so as to compute the TF-IDF of its word terms is not only challenging but it clearly showcases where relevance feedback systems would fail to provide accurate results.

derive the topic labels. As a result, it is very dependent on what terms are in the document. And given the way TF-IDF works for just ten documents, spurious terms can be highly weighted so noise is actually highlighted as topic labels. Furthermore, the results of a TF-IDF compute are single word terms. Topic labels such as "global warming" would appear as two word terms that require an expert to further piece them together. When we consider these limitations, the want for a different solution becomes clear.

3.1 Problem Formulation

Our solution to make Web-based relevance feedback work comes from a simple observation about the relationship between the Web documents, the topic label and the short text, which are all part of the pseudo-relevance feedback process.

Given a tweet t, a human expert could provide a topic label ℓ based on the word terms in t. At the same time, the same word terms from t could be used by the human expert to select a collection of documents $\mathcal{D}_t = \{d_1, d_2, \ldots, d_j\}$, such that these documents also share the topic ℓ. In other words, if all the documents in $\mathcal{D}_t$ are selected for ℓ and that ℓ is some function of t, then ℓ can be seen as a query that returns a set of relevant Web documents $\mathcal{D}_t$. And the query, which is ℓ, is in fact the topic label of t.

The problem in this case is that ℓ is determined by the human expert. For example, the tweet in Figure 2(b) can be labelled by the human expert as $\ell =$'qantas domestic sales'. This would make a good topic label for t and a set of relevant documents to expand the feature space can be easily obtained by searching the Web using the terms from ℓ.

Clearly, the human expert cannot possibly be a component of the relevance feedback system. It would appear that without human expertise to determine ℓ, we won't have a solution. This turns out to be not the case. For a tweet, we often obtain them from a search, a hash tag, or by following another Twitterer. In such situations, we can easily determine the top-level concept $\mathcal{C}$ in relation to the tweet. For example, the tweets in Figure 2 are obtained by searching for 'siggraph' and 'qantas' respectively on Twitter. These query terms are therefore our top-level concepts. As soon as we know $\mathcal{C}$, we can easily derive a set of ℓ-candidates, i.e., $\mathcal{L}(\mathcal{C}) = \{\ell_1, \ell_2, \ldots\}$.

In implementation, one way to easily derive the ℓ-candidates from the top-level concept $\mathcal{C}$ is to use the 'related searches' often suggested by a search engine. For example when $\mathcal{C} =$'qantas', the Bing search engine returns {'frequent flyer', 'international', 'domestic flights', 'staff travel', 'holidays', 'staff credit union', 'flights', 'frequent flyer points account'} as related search topics. If we drill deeper into 'international', we obtain further suggestions which include {'arrivals', 'air fares', 'bookings', 'baggage allowance', etc.}. Clearly, each related search suggestion is a candidate for a topic label. So from $\mathcal{C}$, we can now derive a good set of $\ell-$candidates, i.e., $\mathcal{L}(\mathcal{C})$.

At this point, it becomes clear that each $\ell_i \in \mathcal{L}(\mathcal{C})$ allows us to easily obtain a relevant set of documents $\mathcal{D}_{\ell_i}$. So for each $\ell_i \in \mathcal{L}(\mathcal{C})$, we now have a tuple $\langle \ell_i, \mathcal{D}_{\ell_i} = \{d_i, d_j, \ldots\}\rangle$ or for $\mathcal{L}(\mathcal{C})$, a set of tuples $\{\langle \ell_x, \mathcal{D}_{\ell_x}\rangle, \langle \ell_y, \mathcal{D}_{\ell_y}\rangle, \ldots\}$. To determine the topic label for t obtained via the same concept $\mathcal{C}$, we perform the usual relevance feedback to obtain the tuple $\langle t_\ell, \mathcal{D}_t = \{d_p, d_q, \ldots\}\rangle$, where t_ℓ is the transformed t as per step (i) of a relevance feedback system. Now the solution to our problem of finding a topic label ℓ for t is transformed into finding a tuple in $\{\langle \ell_x, \mathcal{D}_{\ell_x}\rangle, \langle \ell_y, \mathcal{D}_{\ell_y}\rangle, \ldots\}$ where the features in $\mathcal{D}_\ell$ is closest to the features in $\mathcal{D}_t$. The ℓ of this tuple is the topic label for t as their associated documents (or enriched feature space) are the most similar.

By matching features found in $\mathcal{D}_t$ and $\mathcal{D}_\ell$, we are no longer looking for specific word terms. Rather, we are looking for a signature in the set of documents to describe a topic label ℓ. Here, when two sets of documents share a similar signature in their features, we can suggest (or equate) ℓ_t as ℓ. In doing so, the solution of finding ℓ for t is solved.

Algorithm 1 `FindTopicLabel`$(t, \mathcal{C})$

1: build $\mathcal{L}(\mathcal{C})$ from $\mathcal{C}$ using 'related search'
2: obtain $\mathcal{T}_t = \langle t_\ell, \mathcal{D}_t \rangle$ by relevance feedback
3: obtain $\mathcal{T}_{\mathcal{L}(\mathcal{C})} = \{\langle \ell_1, \mathcal{D}_{\ell_1} \rangle, \dots\}$ from search engine
4: **for each** $i \in \mathcal{T}_{\mathcal{L}(\mathcal{C})}$ **do**
5:    // calculate each $\mathcal{S}$ and store result
6:    // in hash table M.
7:    $M(i) \leftarrow \mathcal{S}(\mathcal{S}'(\mathcal{T}_t.\mathcal{D}_t, i.\mathcal{D}_{\ell_x}))$
8: **end for**
9: return $i.\ell \,:\, M(i) > M(j)\, \forall j \neq i$

We can compute the signature in many ways and we present a simple approach in the next section. The strength of the signature approach is that it is a lot more robust against the presence of 'noise' in Web documents. In fact, our approach accepts the presence of noise and incorporates them as part of a topic label's signature.

3.2 Algorithmic Solution

Recall from our earlier discussion, both $\mathcal{D}_\ell$ and $\mathcal{D}_t$ are a set of documents, i.e., $\{d_x, d_y, \dots\}$. The straightforward approach is to take these documents as the respective signature. After all, the combination of Web documents in $\mathcal{D}$ form a collective set of features that describes the topic label.

In this straightforward approach, we can compute a signature similarity score $\mathcal{S}$ to show how similar the signatures are. This is done in two steps: (i) compute the basic cosine similarity between two documents, each drawn from $\mathcal{D}_t$ and $\mathcal{D}_{\ell_x}$ respectively, i.e.,

$$\begin{aligned} \mathcal{S}'(\mathcal{D}_t, \mathcal{D}_{\ell_x}) &= \mathcal{D}_t \times \mathcal{D}_{\ell_x} \\ &= \{\texttt{Sim}(d_i, d_j) : d_i \in \mathcal{D}_t \wedge d_j \in \mathcal{D}_{\ell_x}\} \end{aligned}$$

and then (ii) obtain the average of the cosine similarity scores in $\mathcal{S}'$, i.e.,

$$\mathcal{S}(\mathcal{S}') = \frac{1}{|\mathcal{S}'|} \sum_i s \in \mathcal{S}'$$

The highest signature similarity score $\mathcal{S}$ for an $\ell-$candidate from $\mathcal{L}(\mathcal{C})$ will be selected as the topic label. The algorithm to tie the discussion of our solution together is shown in Algorithm 1.

While the algorithm uses $\mathcal{C}$ to obtain the $\ell-$candidates in Step 1, the solution does not really require it. The presence of $\mathcal{C}$ helps cut the search space, i.e., the number of $\ell-$candidates to consider and consequently, improves runtime performance. Step 2 of the algorithm would be the usual relevance feedback, where t is first transformed into t_ℓ (by the usual preprocessing and POS tagging), and a search conducted using t_ℓ to find a set of relevant documents $\mathcal{D}_t$. In Step 3, the relevant documents for each $\ell-$candidate from $\mathcal{L}(\mathcal{C})$ are retrieved. Again, a good implementation would have cached the frequently used $\ell-$candidates to minimise Web access for performance reasons.

While we didn't implement caching in our prototype, we did limit the size of each document download to 300KB. This greatly improved performance without having to cache any $\ell-$candidate documents, some of which are up to 10MB in our experiments. Empirically, the 300KB performed well without affecting our accuracy. Given that we are only interested in using the documents to form a signature, truncating the download is actually fine.

Steps 4 to 8 simply computes the signature similarity score for each pair of documents in $\mathcal{D}_t$ and $\mathcal{D}_{\ell_x}$ storing the result in a hash table M. Once this is completed, Step 9 returns the ℓ with the top $\mathcal{S}$ score but since one has access to M, the algorithm can return the top-n topic labels as well.

4 Empirical Results

An important aspect of our solution is the premise that a search query is (or will contain) an implicit topic label. This topic label is developed in a search query as users seek relevant documents by refining their search with additional keywords. Over time, this large amount of user queries and clickthroughs has allowed the search engine to learn related searches and the best documents matching each specific query. The indirect consequence of this is that we can now use 'related searches' as a viable source of topic labels based on the solution we presented. It becomes a very powerful way of cutting the search space. At up to two levels deep of related searches, our experimental results show that the topic labels assigned to a tweet will worked very well.

We validated our results as follows. We first obtained a published list of top Twitter queries[1] and hash tags[2] used in 2010 and 2011 respectively. We then performed a Twitter search using these query terms and hash tags to obtain a collection of tweets for our experiment. In this paper, we reported the results from the tweets we collected over the period of July 2012. For each tweet, we recorded the top three and the bottom three topic labels as determined by our algorithm. We then presented the results to a group of Twitter users to assess whether they agree with the topic label assigned.

Our Twitter users were students in a third year software development class taught by one of the authors. Each student was given twenty tweets, half of the tweets were picked from search terms and the other half from hash tags. For each tweet, the top three and bottom three topic labels are shown. The students were to give a score between 1 to 5 to indicate whether they think the top topic label is the best among the six shown. A score of 5 indicates that they fully agree with the algorithm's assessment.

There was a total of twenty students who took part in the assessment. After they made their assessment, we assessed the inter-rater agreement for each tweet across the twenty raters using Flesis's Kappa measure (Fleiss 1977). The Kappa measure is a statistical method to determine the realibility of agreement between raters. In our experiment, the score of 1 to 5 is treated as a nominal measure rather than a ordinal one. Over the twenty tweets, the Kappa value we obtained was just over 0.6 but less than 0.61 (0.6036 to be exact). This places us somewhere between "moderate agreement" and "substantial agreement" according to (Landis and Koch 1977).

Our personal and possibly subjective assessment however motivated us to look deeper into the results as we anticipated a score that clearly puts us in the "substantial agreement" category. We note that the wider the range of scores, the weaker the final result. When we reduced the scoring system to just 'yes', 'no' and 'possibly', the same twenty tweets achieved a better score of 0.73 putting it clearly in the "substantial

[1] `http://blog.sfgate.com/techchron/2010/12/13/gulf-oil-spill-world-cup-top-twitter-trends-for-2010/`

[2] `http://tallskinnykiwi.typepad.com/tallskinnykiwi/2011/12/egypt-the-top-twitter-hashtag-for-2011.html`

agreement" category. We did however have one variable: we had a different group of students to score the same twenty tweets. So while Flesis's Kappa measure provided some statistical validation required for our experiments, we conclude that the best assessment is for the reader to determine the results themselves.

Table 1 shows the tweets we retrieved in July 2012 using the top query terms reported for 2010. The original tweet is shown along with the top/bottom three topic labels (and their $\mathcal{S}$ scores). Table 2 on the other hand shows the tweets we retrieved using the top hash tags reported for 2011. The results are presented in the same way as Table 1. We have given a rather comprehensive list of the results for the readers to make their evaluation. At the same time, we also encourage the reader to download the prototype to test it with their own data. The prototype can be downloaded from `http://www.deakin.edu.au/~leong/getTopic`.

5 Related Works

Topic detection has always been an on-going research question, with reference to the research question from as early as 1996 and discussed with greater interest recently by (Young *et. al.* 2004). Much of the research in topic detection started with conventional text documents, for example, news articles drawn from the Reuters-21578[3] or Web pages from the Open Directory project[4], or in newsgroup. Since then, interests in topic detection moved to short texts such as instant messages and SMS as they became popular. Most of the works however were conducted for a conversational model, i.e., an exchange of emails, SMS or instant messages, e.g., in (Dong *et. al.* 2006, Cselle and *et. al.* 2007, Tian *et. al.* 2010). Soon after, the popularity of blogs moved the research to detecting topics for blog posts, e.g., (Zhang *et. al.* 2011, Xu and Oard 2011). As short texts become increasingly common, e.g., status updates and tweets, the research focus once again shifted with works from (AlSumait *et. al.* 2008, Karandikar 2010, Phuvipadawat and Murata 2010, Cataldi *et.al.* 2010, Zhang and Fan and Chen 2011) being good exemplars.

Among these exemplars, (Cataldi *et.al.* 2010)'s work for example, looks at detecting emerging topics for tweets. Their method begins by modelling tweet content as a feature vector where its word terms are then weighted over time against other tweets drawn from a top-level concept. The idea is that terms with a bigger weight becomes candidates for emerging topics. To confirm a candidate as an emerging topic, user authority and content age are considered. Finally, either a supervised or unsupervised selection algorithm is used to pick word terms that qualify as emerging topics. Therefore, while the objective is to detect a topic label for a tweet, the direction is different. Our goal is to detect a topic for a given tweet. (Cataldi *et.al.* 2010)'s method however requires a constant stream of tweets and requires a window before any emerging topics can be reported.

Most recently in (Zhang and Fan and Chen 2011), the problem of detecting topics from chinese short texts was investigated. The authors approached their research by asking two questions: (i) how to determine the keywords (akin to our topics) in the short text; and (ii) how to expand the keywords to track other short texts that have the same 'topic' but used different word terms. Their work interests us because of their method of finding keywords and then expanding them using hyponymies, i.e., a 'type of' relationship between word terms. Thismay be a way for us to expand our top level concept $\mathcal{C}$ without the need to perform a related search. However, how to relate each expanded keyword to a corpus of documents/short texts isn't immediately obvious.

[3] `http://www.daviddlewis.com/resources/testcollections/reuters21578/`

[4] `http://dmoz.org/`

6 Conclusions

Making sense of short texts is an important research problem as they are becoming increasingly prevalent and ubiquitous. A crucial component to process short texts is the need to know its class or topic label. However, short texts have little features and collectively, has a sparse feature space that makes processing them using conventional algorithms difficult. We present a method to detect topic labels for short texts such as tweets. Our method does not require priori training but produce results that agree well under expert assessment. More importantly, we present a solution that allows the Web to be used as the relevance feedback source. In doing so, our system is guaranteed to be up to date in learning new topic labels. This is crucial in dealing with evolving topics from the large volume of short texts been generated everyday, such as those seen in Twitter.

References

Fleiss, J. L. (1971) "Measuring Nominal Scale Agreement Among Many Raters." Psychological Bulletin, Vol. 76, No. 5 pp. 378 – 382.

Landis, J. R. and Koch, G. G. (1977) "The Measurement of Observer Agreement for Categorical Data." Biometrics. Vol. 33, pp. 159 – 174.

Zhang, C., Fan, X. and Chen, X. (2011) "Hot Topic Detection on Chinese Short Text." Springer Berlin Heidelberg, Vol. 176, pp. 207 – 212.

Lloret, E. (2009) "Topic Detection and Segmentation in Automatic Text Summarization. `http://www.dlsi.ua.es/~elloret/publications/SumTopics.pdf`.

Cataldi, M., Di Caro, L. and Schifanella, C. (2010) "Emerging topic detection on Twitter based on temporal and social terms evaluation." The 10th International Workshop on Multimedia Data Mining. Washington, D.C., ACM, New York, USA. pp. 1 - 10.

Tian, Y., Wang, W., Wang, X., Rao, J., Chen, C and Ma, J. (2010) "Topic Detection and Organization of Mobile Text Messages." The 19th ACM International Conference on Information and Knowledge Management. Toronto, ON, Canada. ACM. New York, NY, USA. pp. 1877–1880.

Dong, H., Hui, S.C. and He, Y. (2006) "Structural Analysis of Chat Messages for Topic Detection." Online Information Review, Vol. 30, pp.496–516.

Perez-Tellez, F., Pinto, D., Cardiff, J. and Rosso, P. (2010) "Clustering Weblogs on the Basis of a Topic Detection Method." The 2nd Mexican conference on Pattern recognition: Advances in pattern recognition.Puebla, Mexico. Springer-Verlag. Berlin, Heidelberg. pp. 342–351.

Moens, M.F. and De Busser, R. (2001) “Generic Topic Segmentation of Document Texts.” The 24th Annual International ACM SIGIR conference on Research and Development in Information Retrieval. New Orleans, Louisiana, United States, ACM, New York, USA. pp.418–419.

Cselle, G., Albrecht, K. and Wattenhofer, R. (2007) “BuzzTrack: Topic Detection and Tracking in Email.” The 12th International Conference on Intelligent User Interfaces. Honolulu, Hawaii, USA, ACM, New York, USA. pp. 190–197.

Phuvipadawat, S. and Murata, T. (2010) “Breaking News Detection and Tracking in Twitter.” The 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology. IEEE Computer Society, Washington, DC, USA. pp.120–123.

AlSumait, L., Barbar, D. and Domeniconi, C. (2008) “On-line LDA: Adaptive Topic Models for Mining Text Streams with Applications to Topic Detection and Tracking.” The 8th IEEE International Conference on Data Mining.IEEE Computer Society, Washington, DC, USA. pp. 190–197.

Xu, T. and Oard, D.W. (2011) “Wikipedia-based Topic Clustering for Microblogs.” Wiley Subscription Services, Inc., Vol. 48, pp. 1–10.

Karandikar, A. (2011) “Clustering short Status Messages : a Topic Model based Approach.” published PhD thesis, The University of Maryland.

Manning, C.D., Raghavan, P. and Schtze, H. (2008) “Introduction to Information Retrieval.” Cambridge University Press.New York, NY, USA.

Blei, D.M., Ng, A.Y. and Jordan, M.I. (2003) “Latent Dirichlet Allocation.”, JMLR.org, Vol. 3, pp.993–1022.

Kristina, T. and Christopher, M. (2000) “Enriching the Knowledge Sources used in a Maximum Entropy Part-of-Speech tagger.” The 2000 Joint SIGDAT Conference: Empirical Methods in NLP and Very Large Corpora.

Bernstein, M.S., Suh, B., Hong, L., Chen, J.,Kairam, S. and Chi, E.H. (2010) “Eddi: Interactive Topic-based Browsing of Social Status Streams.” The 23rd annual ACM symposium on User Interface Software and Technology. New York, USA. ACM, New York, USA. pp. 303–312.

Hulth, A. (2010) “Improved Automatic Keyword Extraction Given More Linguistic Knowledge.” The 2003 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Stroudsburg, PA, USA. pp. 216–223.

Joachims, T. (1998). “Text Categorization with Support Vector Machines: Learn-ing with Many Relevant Features.” In European Conference on Machine Learning (ECML). Springer, Berlin, Germany, pp. 137–142.

Young, W. S., Sycara, K. (2004). “Text Clustering for Topic Detection.” Technical Report (CMU-RI-TR-04-03), Robotics Institute, Carnegie Mellon University.

Zhang, J., Xia, Y., Ma, B. and Yao, J. (2011) “Thread Cleaning and Merging for Microblog Topic Detection.” IJCNLP, 17 December 2011

Bendersky, M. and Croft, W.B. (2010) “Discovering key concepts in verbose queries.” The 31st annual international ACM SIGIR conference on Research and development in information retrieval. Singapore, Singapore. ACM, New York, USA. pp. 491–498.

Table 1: *Tweets and their assigned topic labels obtained by using top query terms reported for 2010.*

Concept $\mathcal{C}$	Tweet	Topic Labels (top/bottom 3 tweets)	$\mathcal{S}$ Score
Gulf Oil Spill	bp you've a lot to answer for! do my eyes deceive me? a captain's view of dolphin health in the gulf http://huff.to/lqy3ya via @huffpostgreen	Gulf Coast Oil Spill Timeline	0.0620
		Gulf Oil Spill	0.0601
		BP Gulf Oil Spill	0.0600
		Gulf of Mexico Map	0.0149
		Gulf Oil Logo	0.0195
		Oil Spill Clean Up Products	0.0214
	photographs of animal skeletons inspired by the gulf oil spill #photography http://bit.ly/leyl3d	Animals Affected by Oil Spills	0.0256
		Animals in Oil Spills	0.0230
		Animals After Oil Spill	0.0226
		Gulf of Mexico Map	0.0065
		BP Oil Spill	0.0081
		Gulf Coast Oil Spill Information	0.0085
	cdc response to the gulf of mexico oil spill http://tinyurl.com/432odhm	Oil Spill Response	0.0719
		Gulf of Mexico Oil Spill	0.0665
		Gulf Coast Oil Spill Information	0.0570
		Lucas Oil Company History	0.0152
		Gulf of Mexico Map	0.0178
		Gulf Coast	0.0200
Inception	what is the most resilient parasite? bacteria? a virus? an intestinal worm? an idea. leonardo dicaprio inception 2010	Inception Review Ending	0.0657
		Inception Film Review	0.0628
		Inception Explanation Ending	0.0600
		Limitless Torrent	0.0095
		Origin of Hooky	0.0119
		Inception Torrent Kick-Ass	0.0129
	inception is one of the sickest, deepest movies ever	Inception the Movie	0.1487
		Inception Movie	0.1485
		Inception	0.1390
		Origins of Islam	0.0063
		Origin of Hooky	0.0063
		Origins of Words	0.0065
	#meta #inception rt @loudboos: seriously, people rt this? rt @jaketapper: ???	Inception Review	0.0324
		Inception Wiki	0.0294
		Inception Reviews	0.0276
		Limitless Torrent	0.0083
		Origin of Hooky	0.0087
		Inception the Movie	0.0097
Haiti Earthquake	powered by action in action in response to the haiti earthquake. see the video - http://bit.ly/yjsoph	Haiti Earthquake Relief	0.0677
		Haiti Earthquake Relief Charities	0.0606
		Haiti Earthquake Relief Red Cross	0.0587
		Avg. Weather for Dominican Rep.	0.0102
		Mermaid Found in Haiti Pictures	0.0119
		Television National d'Haiti	0.0127
	even before the earthquake , conditions in haiti were quite desperate. just behind our hotel in port- http://pinterest.com/pin/235102043018196777/	Earthquake in Haiti CNN	0.0778
		CNN News Haiti	0.0680
		The Earthquake That Hit Haiti	0.0638
		Television National d'Haiti	0.0098
		Haiti TV	0.0109
		Metropole Haiti	0.0110
	update: did haarp cause the earthquake in haiti? http://bit.ly/npmjjd	The Earthquake That Hit Haiti	0.0814
		Earthquake in Haiti 2010 Article	0.0745
		Date Haiti Earthquake Hit	0.0731
		Haiti TV	0.0091
		Television National d'Haiti	0.0102
		Haiti Radio	0.0105

Concept $\mathcal{C}$	Tweet	Topic Labels (top/bottom 3 tweets)	$\mathcal{S}$ Score
Vuvuzela	mind, some love blowing their own trumpet rt @p45c4l linking your twitter account to linkedin is like bringing a vuvuzela to a job interview	South Africa Horn	0.0278
		Horns at World Cup	0.0249
		World Cup Noise	0.0240
		Vuvuzela Hero	0.0035
		YouTube Vuvuzela Alpha Blondi	0.0045
		Vuvuzela Video	0.0057
	cek linkedin rt @15june: rt @p45c4l: linking your twitter account to linkedin is like bringing a vuvuzela to a job interview.	Buy World Cup Vuvuzela	0.0168
		World Cup Vuvuzela	0.0164
		Soccer Horn Vuvuzela	0.0162
		Vuvuzela Hero	0.0027
		YouTube Vuvuzela Alpha Blondi	0.0034
		Vuvuzela Video	0.0046
	#loveprotest time: 10:00am where: uhuru park-freedom corner dress code: kenyan colours,carry a vuvuzela	Horns at World Cup	0.0236
		World Cup Noise	0.0230
		South Africa Horn	0.0217
		Vuvuzela Hero	0.0029
		Vuvu Hero	0.0053
		YouTube Vuvuzela Alpha Blondi	0.0058
Apple iPad	google's nexus 7 could force apple's hand on 'ipad mini' -	iPad Mini 2012	0.0677
		Mini iPad	0.0654
		New Tablets	0.0649
		Apple	0.0096
		AT&T Wi-Fi	0.0106
		Apple iPhone Support	0.0117
	microsoft surface vs apple new ipad http://bit.ly/mywxqj	New Tablets	0.0676
		HP iPad-like	0.0628
		HP iPad Computer	0.0625
		Apple	0.0108
		AT&T Wi-Fi	0.0116
		Apple iPhone Manual	0.0121
	google unveils $199 tablet to take on ipad - http://interaksyon.com http ://fb.me/1aw2h68p7	New Tablets	0.0675
		iPad 2 Price Drop	0.0608
		HP iPad On Sale	0.0602
		AT&T Wi-Fi	0.0096
		Apple	0.0135
		Apple iPhone Support	0.0158
Google Android	google's new youtube app for android 4.0 is rolling out today http://tnw.to/n0dj by @harrisonweber	Google Android M Downloads	0.0805
		Google AppBrain	0.0691
		Google Market Download	0.0682
		Transaction Fees	0.0098
		What Does Apps Mean	0.0153
		Google Plus Post	0.0156
	google's new android 4.1 jelly bean os detailed http://bit.ly/n5p5up	Android Ice Cream Sandwich	0.0909
		Ice Cream Sandwich Operating System	0.0772
		Ice Cream Sandwich Android Release	0.0731
		Transaction Fees	0.0178
		What Does Apps Mean	0.0180
		Google Android M Downloads	0.0192
	google nexus 7 is official, shows off android 4.1 jelly bean http://cnet.co /oxgw6m	Ice Cream Sandwich Tablets	0.0707
		Ice Cream Sandwich Android Tablet	0.0704
		Android Tablets 2011	0.0701
		What Does Apps Mean	0.0156
		Transaction Fees	0.0175
		Live Android Downloads	0.0207

Concept $\mathcal{C}$	Tweet	Topic Labels (top/bottom 3 tweets)	$\mathcal{S}$ Score
Justin Bieber	one direction will be the biggest boyband in the world by the end of this year. - justin bieber	Selena Gomez and Justin Bieber	0.0422
		Justin Bieber Paternity Suit	0.0404
		J.B. Selena Gomez Pregnant	0.0402
		YouTube Videos	0.0140
		Project Live Love	0.0152
		Countdown to 18th Birthday	0.0153
	official: justin bieber's 'believe' is year's biggest debut, bows at no. 1 - http://bit.ly/mtlgfu	How Old Is Justin Bieber	0.0428
		YouTube J.B. Music Videos	0.0395
		Justin Bieber Lyrics	0.0389
		Project Live Love	0.0089
		Happy Birthday 18th	0.0097
		YouTube Videos	0.0101
	niall horan in justin bieber's boyfriend video. http://pic.twitter.com/edcgvwva	Selena Gomez Justin Bieber Kiss	0.0438
		YouTube J.B. Baby Baby	0.0403
		YouTube J.B. Favorite Girl	0.0403
		Project Live Love	0.0120
		Happy Birthday 18th	0.0131
		Countdown to 18th Birthday	0.0141
Harry Potter & the Deathly Hallows	when a muggle saw me reading the deathly hallows book, he asked me "how does harry potter end?" i simply answered "it doesn't."	H.P. SparkNotes Sorcerer's Stone	0.0541
		Harry Potter Reviews	0.0476
		Hogwarts Professor Names	0.0470
		Dumbledore's Army Font	0.0114
		The Wizard Stone	0.0135
		Harry Potter Fun and Games	0.0146
	harry potter and the deadly hallows, part 1 (four-disc blu -ray deluxe edition): the 4-disc ultimate blu-ray edit... http://amzn.to/obfbrt	Harry Potter Reviews	0.0532
		Harry Potter Actors	0.0481
		Harry Potter Film Cast	0.0440
		Dumbledore's Army Font	0.0093
		Harry Potter Fun and Games	0.0119
		Staff Trivia Questions	0.0122
	rt if you cried throughout most of harry potter and the deathly hallows part 2.	Deathly Hallows Movies	0.1217
		Deathly Hallows Official Site	0.1183
		H.P. and the Deathly Hallows	0.1087
		Staff Trivia Questions	0.0060
		The Wizard Stone	0.0083
		Actor Killed Today	0.0097
Pulpo Paul	-hola, c mo te llamas? -yogi, y t ? -paul. - jajaja!, no mames como el pulpo....	Preguntar Al Pulpo Paul	0.0186
		Spanish Octopus Recipes	0.0092
		Spanish Octopus Tapas	0.0091
		Stoneware Drinking Glass	0.0027
		Bell Co51 Octopus Cup Holder	0.0031
		Al Paul Car Wash	0.0032
	i have doubts about today s spain match but if @virginiecapric (the new pulpo paul) says germany - spain, well,here we go to the final!!	Octopus World Cup Prediction	0.0513
		Paul the Octopus Predictions	0.0505
		Paul the Psychic Octopus	0.0485
		What Is Pulpo	0.0064
		Al Paul Car Wash	0.0075
		Stoneware Drinking Glass	0.0088
	paul the octopus is dead actually so im guessing el pulpo ra l too	Preguntar Al Pulpo Paul	0.0374
		Pulpo Recipe	0.0189
		Pulpo Gallego Recipe	0.0177
		Al Paul Car Wash	0.0023
		Bell Co51 Octopus Cup Holder	0.0035
		Make Your Own Coolie Cup	0.0039

Table 2: *Tweets and their assigned topic labels obtained by using top hash tags reported for 2011.*

Concept $\mathcal{C}$	Tweet	Topic Labels (top/bottom 3 tweets)	$\mathcal{S}$ Score
egypt	another horrific attack on a woman in cairo http://on.cnn.com/lw2hbq #egypt #tahrir	Clashes Egypt	0.0894
		Egypt Virginity Test	0.0752
		Egypt Soccer Game Deaths	0.0719
		Quiz On Middle East	0.0043
		Soccer Game Cup	0.0086
		Greek Gods	0.0114
	#egypt ex-oil min sameh fahmy + hussein salem get 15 yrs: 'squandering public funds' in #israel gas deal http://tinyurl.com/6wdd8sq	Soccer Game Cup	0.0082
		70 Dead Soccer	0.0072
		Pyramid	0.0071
		Egypt God Horus	0.0010
		Proof of Virginity	0.0011
		Map Africa	0.0011
	christians nervous under new president in egypt. http://bit.ly/lvwza0	Clashes Egypt	0.0745
		Egyptian Soccer Riot	0.0526
		Egypt Soccer Game Deaths	0.0485
		Quiz On Middle East	0.0048
		Soccer Game Cup	0.0087
		Greek Gods	0.0088
tigerblood	charlie sheen calls tmz to address hotel lies about him partying okay, we believe you charlie. http://ow.ly/bsppi #tigerblood	Tiger Blood Quote	0.0569
		Charlie Sheen Drinking Tiger Blood	0.0550
		Charlie Sheen Tiger Blood Interview	0.0528
		Paula Deen Riding a Bunchie	0.0061
		Tiger Blood Snow Cone	0.0066
		Alex Pardee T-Shirts	0.0078
	i know charlie sheen aint cool anymore but i still got #tigerblood and im still #winning	Tiger Blood Intern	0.0258
		I Got Tiger Blood	0.0244
		Charlie Sheen Tiger Blood Video	0.0226
		Tiger Blood Snow Cone	0.0029
		Tiger Blood Snow Cone Syrup	0.0033
		Tiger Pharmacy Steroids	0.0050
	power - kanye west is such a good song omg	Tiger Blood Quote	0.0224
		Charlie Sheen Tiger Blood Interview	0.0222
		Charlie Sheen Tiger Blood Comment	0.0211
		Charlie Sheen Tiger Blood Shirt	0.0033
		Tiger Blood Snow Cone	0.0052
		Paula Deen Riding a Bunchie	0.0057
threewordstoliveby	#threewordstoliveby love your life (:	Great Quotes to Live By	0.0236
		Quotes to Live by Tumblr	0.0226
		Great Words to Live By	0.0213
		Lyrics2liveby	0.0014
		Lyrics 2	0.0038
		Lyrics to Live By	0.0040
	#threewordstoliveby loyalty is everything	Great Quotes to Live By	0.0289
		Best Words to Live By	0.0254
		Shook Ones Part 2 Lyrics	0.0250
		Lyrics2liveby	0.0018
		Lyrics to Live By	0.0038
		Tumblr Lyrics to Live By	0.0050
	#threewordstoliveby faith , love, hope	Great Quotes to Live By	0.0304
		Great Words to Live By	0.0273
		Morning Quotes to Live By	0.0259
		Lyrics2liveby	0.0013
		Lyrics 2	0.0039
		Lyrics to Live By	0.0045

Concept $\mathcal{C}$	Tweet	Topic Labels (top/bottom 3 tweets)	$\mathcal{S}$ Score
japan	mexico s olympic squad to play friendly v le n on july 5 + will face the england, spain and japan olympic squads prior to london olympics.	World Cup Football Japan	0.0327
		USA Japan Game	0.0304
		Japan US Women Soccer	0.0290
		Soft On Demand Sod	0.0038
		SOD Create	0.0040
		Princess of China Lyrics	0.0052
	the japan night life! all of the lights http://instagr.am/p/maxfdcyda6/	Population of Tokyo	0.0250
		Population of Japan	0.0228
		USA Japan Game	0.0210
		Soft On Demand Sod	0.0044
		China Anne McClain	0.0046
		Princess of China Lyrics	0.0064
	kim soo hyun to head for japan to promote moon that embraces the sun! http://bit.ly/kfrocr	Japan Earthquake Anniversary	0.0325
		2011 Japan Earthquake	0.0312
		Earthquake Japan 2012	0.0308
		Princess of China Lyrics	0.0043
		China Anne McClain	0.0047
		Soft On Demand Sod	0.0056
superbowl	jets fans this man has been working out. look at those arms. with him and sanchez u heard it here first superbowl http://pic.twitter.com/c4xynzmi	Super Bowl Odds	0.0356
		Super Bowl 44 Odds	0.0320
		Super Bowl Scores 2012	0.0308
		CBS Local Chicago	0.0023
		2012 Calendar	0.0069
		2012 Predictions	0.0136
	the supreme court are those dudes who did "superbowl shuffle", right?	Super Bowl 2012 New Orleans	0.0216
		2012 Predictions	0.0213
		Super Bowl 2014	0.0205
		Super Bowl 43	0.0058
		CBS Local Chicago	0.0062
		superbowl	0.0073
	breaking: cnn reports the indianapolis colts have won the super bowl.	Super Bowl 2014	0.0121
		Halftime Show Super Bowl 2012	0.0109
		Where Is Super Bowl 2016	0.0106
		CBS Local Chicago	0.0009
		Super Bowl 44 Logo	0.0018
		Prince Halftime Show Super Bowl	0.0024
jan25	martyr: ahmed hashim el-sayyed age 25 died in #alex on 28jan #egypt #jan25	Egyptian Revolution of 1952	0.0320
		Day of Rage Egypt	0.0310
		Revolution Egyptian	0.0301
		DirecTV Revolution 2012	0.0074
		Egyptian Revolution 2011 Photos	0.0083
		Lending in Bank	0.0086
	martyr: omar fathi nour al-barbari died in maadi on jan28 by family's received his body ...(more) http://bit.ly/lre59j #egypt #jan25	Egyptian Revolution of 1952	0.0435
		Day of Rage Egypt	0.0422
		Revolution Egyptian	0.0408
		DirecTV Revolution 2012	0.0070
		Lending in Bank	0.0083
		Egyptian Revolution 2011 Photos	0.0087
	martyr: aly elnabawy age 55 died in ismailia by gunshots ..., fisher #egypt #jan25	Day of Rage Egypt	0.0681
		Egyptian Revolution of 1952	0.0628
		Revolution Egyptian	0.0621
		Egyptian Revolution 2011 Photos	0.0151
		Lending in Bank	0.0158
		25-Jan	0.0159