Predicting Semantic Annotations on the Real-Time Web*

Elham Khabiri
Texas A&M University
College Station, TX
khabiri@cse.tamu.edu

James Caverlee
Texas A&M University
College Station, TX 77843
caverlee@cse.tamu.edu

Krishna Y. Kamath
Texas A&M University
College Station, TX 77843
kykamath@cse.tamu.edu

ABSTRACT

The explosion of the real-time web has spurred a growing need for new methods to organize, monitor, and distill relevant information from these large-scale social streams. One especially encouraging development is the self-curation of the real-time web via *user-driven linking*, in which users annotate their own status updates with lightweight semantic annotations – or *hashtags*. Unfortunately, there is evidence that hashtag growth is not keeping pace with the growth of the overall real-time web. In a random sample of 3 million tweets, we find that only 10.2% contain at least one hashtag. Hence, in this paper we explore the possibility of predicting hashtags for un-annotated status updates. Toward this end, we propose and evaluate a graph-based prediction framework. Three of the unique features of the approach are: (i) a path aggregation technique for scoring the closeness of terms and hashtags in the graph; (ii) pivot term selection, for identifying high value terms in status updates; and (iii) a dynamic sliding window for recommending hashtags reflecting the current status of the real-time web. Experimentally we find encouraging results in comparison with Bayesian and data mining-based approaches.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

Keywords

social media, hashtag prediction

1. INTRODUCTION

The real-time web has grown at an astonishing rate in the past several years. As one example, Twitter has rapidly grown from handling 5,000 tweets per day in 2007 to 50 million tweets per day in 2010 to 140 million per tweets per day in 2011. At an order of magnitude higher, Facebook reported in 2009 that it was handling around 1 billion chat messages per day,[1] and there is widespread evidence of massive growth in web-based commenting systems (like on Reddit, Digg, and NYTimes) and other real-time "social awareness streams" [17].

Figure 1: Two sample tweets annotated with the hashtag #health.

Coupled with this explosion in content reflecting the real-time interests of web users is the need for new methods to organize, monitor, and distill relevant information from these large-scale social streams. Along this line, there have been a number of recent efforts aimed at providing a search and analytics tools over the real-time web for making sense of the aggregate activities of millions of users [2, 3, 12, 24]. One especially encouraging development is the self-curation of the real-time web via *user-driven linking*, in which users annotate their own status updates with lightweight semantic annotations – or *hashtags*. On Twitter, for example, these hashtags are inserted into tweets by users and serve many functions. For example, some reflect categorical information about the tweet as in Figure 1, where both have been annotated with the hashtag #health. Some hashtags reflect events related to a tweet (e.g., #ht2012) and many others reflect the sentiment of the tweet (e.g., #Iloveapple, #sucks). And of course, as user-generated descriptors, some are nonsensical or of interest only to the user posting the hashtag.

By linking status updates to hashtag-like semantic descriptors, users provide a potentially scalable mechanism to organize the real-time web as it continues to grow. As users continue to post status updates with hashtags, there will always be additional semantic cues for organizing these updates. For example, as new issues become associated

*This work was supported in part by DARPA grant N66001-10-1-4044 and by a Google Research Award. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsors.

HT'12, June 25–28, 2012, Milwaukee, Wisconsin, USA.

[1] `http://www.facebook.com/note.php?note_id=91351698919`

with the "Health" concept, we would expect to see new updates using the #health hashtag. In this way, the user-driven semantic annotation of the real-time web could provide an evolving framework for improving information navigation in these systems (by linking similar updates according to common hashtags), by inducing concept hierarchies over these status updates (so that #cancer-related updates are organized under the umbrella of #health), for supporting serendipitous exploration of the real-time web, improving the recall of search operators (by returning both #apple and #mac related updates for queries about the company), and so on. Indeed, a recent study of Twitter search shows that hashtags are popular as queries, and that these queries are often repeated so that users may monitor search results [25]. By linking untagged updates with hashtag-like semantic descriptors, such searches could have expanded coverage.

Unfortunately, there is evidence that hashtag growth is not keeping pace with the growth of the overall real-time web. In a random sample of 3 million tweets, we find that only 10.2% contain at least one hashtag, meaning that 89.8% are un-labeled and would be left out of any hashtag-oriented search or monitoring application. In addition, there is mounting evidence that many hashtags may convey little semantic information or are being used as tools of spammers and other polluters of these systems [10, 13, 14]. Hence, in this paper we explore the possibility of predicting hashtags for un-annotated status updates. Can we determine the appropriate semantic label for an update?

Toward this end, we propose and evaluate a graph-based prediction framework in which terms in status updates are linked to hashtags based on their co-occurrence. Since many relevant hashtags may not co-occur with all possible terms, we develop a path aggregation technique for scoring the closeness of terms and hashtags in the graph. In this way, high-value hashtags may be associated with status updates, even if no terms in the update have ever co-occurred with the hashtag. Additionally, we augment the baseline method with a pivot term selection approach for identifying high value terms in status updates, and a dynamic sliding window for recommending hashtags reflecting the current status of the real-time web. Experimentally we find encouraging results in comparison with Bayesian and data mining-based approaches.

The organization of the rest of the paper is as follows. Section 2 gives a brief overview of the related work. In Section 3 we provide the formal definition of the problem of predicting semantic annotations for the real-time web and present the step-by-step development of the proposed graph-based hashtag prediction framework. Section 4 details the Twitter-based dataset, introduces several alternative approaches that we compare the proposed model with, and presents the results of a comparative evaluation of the proposed approach. Finally in Section 5 we conclude with a summary and some final thoughts.

2. RELATED WORK

User-driven tagging is one of the organizing principles of most social media services – including image tagging (e.g., on Flickr), video tagging (e.g., on YouTube), and web page tagging (e.g., using Del.ico.us), among many others. And in these contexts, there has been considerable work in recommending tags. In one direction, researchers have sought methods to aggregate the collective knowledge of web users

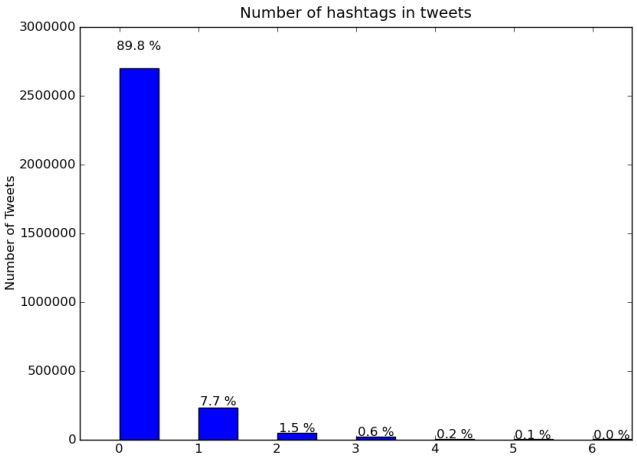


Figure 2: Most tweets are annotated with no hashtags. In a random sample of 3 million tweets, we find that 7.7% contain exactly one hashtag, and 2.5% contain more than one hashtag.

to expand the small set of tags applied to a resource with other user-contributed tags [20, 6]. In a different direction, other studies have recommended personalized tags for each user based on the user's history, bookmarks, and other personal documents [5, 18]. In collaborative filtering based approaches [20, 22, 27, 7, 16], the number and frequency of tag co-occurrence builds the core model of tag recommendation. Given a set of tags already input by the user for a new resource such as a picture, URL, or a blog post, these algorithms suggest new tags based on the number of co-occurrences of such input tags with the previous annotations. There are many studies on the usage of data mining approaches (like association rules) to predict the appropriate tags for content-rich resources such as webpages [8][26]. Several efforts have focused on graph-based approaches, in which the relationships among tags, resources and users are modeled as a tri-partite [11] graph. In such settings, important "power" tags, users, and resources may be identified through the application of a PageRank-like iterative algorithm. Similarly a tag-document bipartite graph has been used as the basis to cluster tags and documents, as discussed in [21]. In addition to these efforts, there have been many other approaches for tag recommendation [15, 9, 4, 28, 30].

Compared to these efforts, predicting semantic annotations for the real-time web differs in three fundamental ways. First, in traditional social media tag prediction, the tagged resource itself (e.g., the video, the image) is typically made available for collaborative tagging. That is, an image on Flickr may attract dozens or hundreds of contributors who provide their own tagging perspective on what the image is, providing a rich source of tagged information for a single image. In comparison, a status update on the real-time web is annotated by just one user and typically with only one hashtag, meaning there is not a rich collection of collaboratively shared hashtags available to describe a single status update. Second, for the purposes of hashtag prediction, the status update itself is a sparsely described object.

Most status updates are short (as on Twitter, where there is a 140 character limit) and so there is little evidence in the status update itself; in contrast, web pages and other social media often contain richly available descriptive evidence (e.g., in the text of the page itself) to augment tag prediction. Third, the real-time web is necessarily a rapidly evolving medium, with millions of updates per day and highly-dynamic tagging behavior, meaning that the tags themselves may rapidly evolve and change in use and purpose (as compared to a Flickr photo of a well-known landmark, in which the tags associated with the landmark are typically much longer-lived and less dynamic). Hence, it is important to develop a new approach for predicting semantic annotations on the real-time web.

3. PREDICTING SEMANTIC ANNOTATIONS

In this section, we formalize the problem of predicting semantic annotations for the real-time web and introduce a hashtag graph-based prediction framework.

3.1 Problem Statement

Let $T = \{T_1, T_2, ..., T_n\}$ be the set of status updates (i.e., tweets), and $T_i = \{u_1, u_2, ..., u_m\}$ be a set of unigram terms, and $H = \{h_1, h_2, ..., h_m\}$ be the set of hashtags. Our goal is for an unlabeled status update T_i to predict a hashtag h_j that "correctly" annotates the update. Of course, it is challenging to determine what is the "correct" choice of hashtag. In one direction, the evaluation of hashtag prediction can be based on a user study in which human subjects are asked to evaluate the quality of predicted hashtags for each of the testing tweets. A recent study [6] argues that human evaluation of tags may lead to errors in assessment due to multi-lingual tags, missing context, differences of level of details, and the interdependence of tags. Alternatively, we can adopt a purely machine-based evaluation framework in which the prediction model is built over a training set and then used to predict the hashtags for a test set. In this case, the hashtags themselves are removed from the test set and then the quality of the prediction is in identifying the actual hashtag that had been used. Such an approach, while providing less flexibility (e.g., by not accepting #nba as a reasonable tag for a sports-related tweet actually annotated with #basketball), does provide for fast evaluation and comparison across multiple methods. Hence, in this paper, we adopt this second approach.

Concretely, we adopt an evaluation framework in which a portion of the data is used as a training set for learning the prediction model, and a separate testing set is used for evaluation. The model is used to predict the hashtags of test tweets in which all the hashtags are removed. The predicted k tags are denoted t_{pred}. The actual tags applied to the tweet are denoted t_{real}. For varying values of k, we can evaluate the quality of hashtag prediction using precision:

$$prec = \frac{|t_{real} \cap t_{pred}|}{|t_{pred}|}$$

where predicting only hashtags that are actually used results in a precision of 1, whereas predicting none of the correct tags actually used results in a precision of 0. We additionally evaluate the quality of hashtag prediction using recall:

$$rec = \frac{|t_{real} \cap t_{pred}|}{|t_{real}|}$$

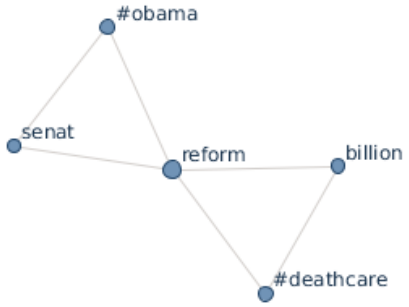


Figure 3: Although "senate" and "#deathcare" have not appeared together in any tweets, the two are related, as revealed by the short path (2 hops) in the semantic graph.

where identifying all of the correct hashtags results in a recall of 1. Finally, we also consider the combined F-measure:

$$f = \frac{2 \times prec \times rec}{prec + rec}$$

We measure the overall precision, recall and F-measure by averaging over all testing tweets.

3.2 Hashtag Graph-Based Prediction

Given the overall goal, we propose in this section a graph-based prediction approach. The core idea is to identify implicit relationships among the hashtags and terms used in tweets to build a semantic graph that may then be used to connect the terms in unlabeled tweets to the appropriate hashtags. The baseline assumption is that terms and hashtags that are used together are related and hence close in terms of meaning. For example, Figure 3 shows a subgraph built over a large Twitter dataset (described more fully in the experimental evaluation) in which a term like "senate" is linked to "reform" due to the use of both terms in many tweets. Similarly, "senate" and the hashtag "#obama" are linked due to their co-occurrence. However, strictly considering co-occurrence alone will miss the implicit connection between "senate" and "#deathcare". Returning to Figure 1 we find that terms like "sick" and "patient" are close in the semantic graph to the hashtag "#health". By identifying these implicit connections across all of the terms used in an unlabeled tweet, the proposed approach seeks to find hashtags that are close in terms of this semantic graph. Hence, for a tweet T, we can estimate the appropriateness of a hashtag h as an aggregation operation over all of the terms occurring in T:

$$score(T, h) = \sum_{t_i \in T} p\text{-}score(t_i, h) \quad (1)$$

where $p\text{-}score(t_i, h)$ is an estimate of $p(h|t_i)$ – the conditional probability of the hashtag being used, when t_i is observed.

However, naive application of such an approach will face several challenges. First, how should evidence from different terms from a single tweet be aggregated to find the consensus of the tweet? In other words, a tweet containing terms like "senate" and "healthcare" may be closely linked to many candidate hashtags. In what ways can we distill the most

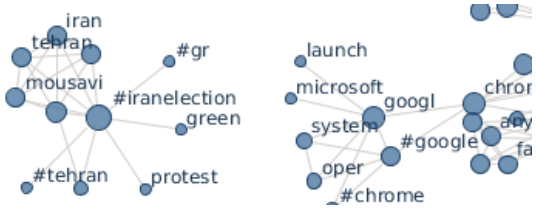


Figure 4: Relationship among hashtag and terms. The left side shows terms and hashtags related to the Iran election; the right side is technology-centric.

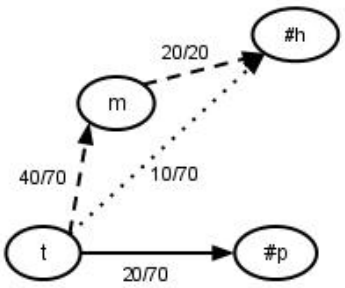


Figure 5: The score of hash #h related to term t is calculated by the summation of all the path scores between these two nodes.

likely hashtags from a long list of candidates? Second, aggregating the evidence across all terms in a tweet may lead to topic drift, in which particular terms are closely linked to hashtags that are not at all relevant to the overall tweet. For example, the term "state"' in the first tweet shown in Figure 1 may be linked to hashtags associated with mental states, states like Texas and Oregon, and other concepts not at all linked to the hashtag "#health". Third, the probability of a hashtag given a term may change over time. For example, the term "obama" will be closely linked with different terms and different hashtags based on the political debate of the day, whether the election is upcoming, and so on. Hence, careful determination of the temporal relationships between terms and hashtags is important.

With these challenges in mind, we now detail three specific steps toward hashtag graph-based prediction: (i) a path aggregation technique for scoring the closeness of terms and hashtags in the graph; (ii) pivot term selection, for identifying high value terms in status updates; and (iii) a dynamic sliding window for recommending hashtags reflecting the current status of the real-time web.

3.2.1 Linking Terms and Hashtags

First, we build a semantic graph and propose a path aggregation technique for scoring the closeness of terms and hashtags in the graph. We build a graph $G = (N, E)$ with nodes $N = \{n_1, n_2, ..., n_m\}$ in which n_i is either a term or a hashtag and edges $E = \{e_1, e_2, ..., e_r\}$ in which e_j is the weighted edge between two nodes. To avoid noise and to keep our graph less polluted we only create an edge between two nodes when the number of co-occurrences is greater than a threshold. The co-occurrence is measured by considering all tweets in the training set and counting the number of times two elements (either terms or hashtags) occur together in the same tweet. In this way, we may filter out non-important edges that have happened by chance. A sample graph is illustrated in Figure 4, which shows the relationship between terms and hashtags.

But what is the appropriate weighting function for edges between nodes? This weighting function can be used to identify the relative "closeness" of terms and hashtags that are directly connected. In one direction, the co-occurrence count itself may be used. Consider the case that terms A and B have co-occurred 10 times together, and both A and B occur across all tweets exactly 10 times each. So, A and B always co-occur together and never apart. Now suppose A appears 100 times, but only in 10 cases did it co-occur with B. In this case, the "closeness" of A and B is less than in the first case. Hence, we normalize the co-occurrence value by the number of the times a term has appeared in the whole corpus which is equal to the number of outlinks of that node. This *normalized weight score*, $NW_{(n,n+1)}$, is the normalized weight of the edge between node n and node $n+1$:

$$NW_{(n,n+1)} = \frac{W_{(n,n+1)}}{\sum_{p \in Outlink(n)} W(n,p)} \quad (2)$$

where $W_{(n,n+1)}$ is the co-occurrence count of the two elements n and $n+1$ (terms or hashtags). By this normalization we consider the amount of the node devotion to the relationship with another node. Therefore an edge to a more general term will receive smaller weight in comparison with an edge to a more specific term. Now the question is how to measure the "closeness" of nodes that are more than one hop away? What will happen if the relevant hashtag was found two or three hops away? Shall we penalize the score of the hashtags that are located farther from a particular term? And to what degree?

To formalize the problem we say that the hashtag h is reachable from term t_i in radius m as $t_i \rightsquigarrow^m h$. For a single path from a term to a hashtag, we propose to consider the product of all the edge weights in the path, where the edge weights themselves are decayed by a factor β. The decay factor is to penalize the nodes that are far from the source node, so that we still consider them as candidate hashtags but with lower significance than ones that are directly connected in the graph. The score for a path is then:

$$score\_path(t_i \rightsquigarrow^m h) = \prod_{n=0}^{m-1} NW_{(n,n+1)} * \beta^n \quad (3)$$

where the normalized edge weights between nodes are decayed, and so the farther a hashtag is from a term source node, the less score it gets.

To find the overall score of a hashtag h from term t_i we measure aggregation scores of all of the paths existing between them. So that if a hashtag is reachable by more than one path it shows more relevance to the term in comparison with the case that it is only reachable by one path. Hence, this aggregated path score is:

$$p\text{-}score(t_i, h) = \sum_{m=1}^{M} score\text{-}path(t_i \rightsquigarrow^m h) \quad (4)$$

where we consider all paths from a term to a hashtag. For example in Figure 5 hashtag h is reachable from a term t

once in 1 hop (the dotted path,) and the other time through 2 hops (the dashed path). In this way, we link terms to hashtags.

3.2.2 Selecting Pivot Terms

Given the semantic graph and the method for linking terms to hashtags, the aggregation method described in Equation 1 can be applied immediately. However, by considering all terms in a tweet for finding appropriate hashtags may introduce noise in the case of spurious term-hashtag connections caused by considering isolated linkages between terms and hashtags without regard for the overall tweet content. For example for the tweet "So there is actually a python module called pyjamas", many of the terms are not significant for predicting an appropriate semantic annotation; "so", "there", "is", and so on are relatively common terms and they convey little information about the tweet. In contrast, "python", "module", and "pyjamas" are all strong cues.

Hence, we propose to select a subset of terms from each tweet based on their high information content. This *pivot term selection* results in keeping the model small and eliminating terms that are ineffective for tag prediction. While there are a number of ways to select pivot terms, we consider two approaches – by inverse document frequency and by entropy.

To select pivot terms by inverse document frequency measure (IDF), we consider the number of times a term was used in all the tweets – df_t – within the training set.

$$IDF(t) = log\frac{N}{df_t} \quad (5)$$

where N is the total number of tweets in the training set. Hence we identify the terms with high IDF and eliminate the more general terms with low value.

For entropy-based pivot term selection, we identify terms with low entropy (which tend to be more specific) and eliminate terms with high entropy (which tend to be more general non-informative terms). The entropy of term t is measured as:

$$Entropy(t) = -\sum_{h_i \in H} p(h_i|t) \times log(p(h_i|t)) \quad (6)$$

where H is the set of all hashtags that co-occur with term t. By selecting as pivot terms those terms that are low in entropy, the goal is to find good predictors of hashtags. To illustrate, Table 1 shows a sample of terms with high entropy versus those with low entropy in a large collection of tweets (described in the following section). The terms with lower entropy are more specific and terms with higher entropy are more general terms.

3.2.3 Sliding Windows

Finally, since the real-time web is constantly changing, we augment the baseline hashtag prediction approach with a *sliding window*. The intuition is that the recency of hashtags is a strong indicator of their appropriateness for annotating tweets. For example, events such as Gaddafi's death or the Super Bowl, shifting user interests, or announcements of new products will drive a changing portfolio of hashtags in use by users of the real-time web. Thus, a higher importance can be assigned to more recent hashtags than those introduced a long time ago.

Concretely, we propose to build the semantic graph based on the past Δ time, rather than considering the entire history. The sliding window could be an hour, day, week, or month. In this way, the predicted hashtags can be based on this sliding window as illustrated in Figure 6, reflecting the current composition of hashtags.

Low Entropy	High Entropy
twade sherlock	win house
vancouver perception	save chance
tweekly intriguing	post prize
wesson legend	good top
naraku equivalent	american person
crunchy irm	end night
tempting tub	tip group
jumper drinking	stop hot
chilli whistle	week nice

Table 1: Sample of terms with high/low entropy.

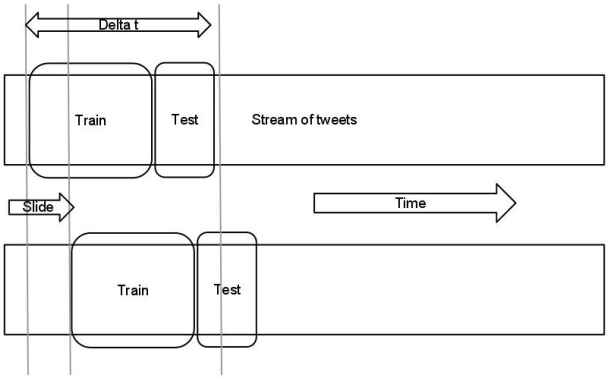


Figure 6: From the stream of tweets we construct a time-window of Δ and split the data into 80-20 train-test sets within each window. We repeat the experiments for all sliding windows.

Additionally, the model may be smoothed by considering a mixture of both the most recent window and the global history. For example if we build the model based on the past day, could we improve it by considering the information from the past week? Therefore we suggest a smoothing process that takes into account both the recent history and the complete history:

$$smooth\text{-}score(t_i, h) = 0.9 * p\text{-}score(t_i, h) + 0.1 * G(h|t_i) \quad (7)$$

where $G(h|t_i)$ is the global probability of hashtag h given term t_i. Also note that the global model can be calculated offline, so that it can be efficiently incorporated into the sliding window approach.

4. EVALUATION

In this section, we present an experimental study of hashtag recommendation. We describe the dataset, metrics used, introduce several alternative adaptations of tag prediction in social media to the problem of hashtag prediction, and present the results of a comparative study.

4.1 Dataset

For the experimental evaluation, we adopt the Stanford Twitter dataset containing 344 million Twitter posts from 20 million users covering a 6 month period from 06/01/2009 to 11/31/2009 [29]. This dataset contains about 20-30% of all public tweets published on Twitter during this time frame. After removing tweets with empty text, we arrive at a dataset described in Table 2. Eliminating terms and hashtags with length less than 2 and those that were used fewer than 10 times in tweets, we arrive at nearly 500K unique terms and 100K unique hashtags in the dataset. We randomly split the data into an 80/20 mix, so that 80% of the tweets with hashtags are used as training and 20% of tweets with hashtags are for testing.

Total number of tweets	344,139,347
Total tweets with hash	36,558,421
Size of term dictionary	502,684
Size of hash dictionary	134,522

Table 2: Statistics of Twitter dataset.

4.2 Alternative Methods

As we discussed in the related work section, there have been a number of studies of tag recommendation over traditional social media. We now describe adaptations of several of these alternative approaches, in which we customize the techniques for hashtag prediction.

4.2.1 Adapting Flickr-Based Tag Recommendation

The first approach was developed in the context of tag recommendation for photos on Flickr [20]. In this context, it is assumed that each photo has already been tagged by some set of users. Based on the co-occurrences of these tags with tags associated with other photos, the method can recommend additional tags for the baseline photo. The authors propose two aggregation methods for scoring and ranking candidate tags in order of their appropriateness – by voting and by summation. The *voting-based method* considers the number of times that a candidate tag was seen. As an example consider A and B as the two input tags for a photo. Suppose A co-occurs with $\{M, N\}$ and B co-occurs with $\{M, P\}$. The votes will be $\{M : 2, N : 1, P : 1\}$, meaning that M will be the most highly-rated new tag to be recommended. The *summation-based method* additionally uses the co-occurrence value of the tags. Suppose for the same example that the co-occurrence values are: $A \rightarrow \{M : 1, N : 9\}$ and $B \rightarrow \{M : 2, P : 10\}$. Then the summation-based method will score the three tags as: $\{M : 3, N : 10, P : 10\}$, where now M is the lowest-score tag. Translating from Flickr tag recommendation to our context, we can consider each term as an object and then consider all of the hashtags that were used with this term across all tweets. Therefore we have each tweet made of p terms: $T_i = \{t_1, t_2, ..., t_p\}$. Hence, the voting-based method becomes:

$$vote(h, T) = \sum_{t_i \in T} vote(h, t_i)$$

where

$$vote(h, t_i) = \begin{cases} 1 & \text{if } h, t_i \text{ co-occur} \\ 0 & \text{otherwise} \end{cases}$$

in which we consider all of the hashtags that have co-occurred with each of the terms in tweet T and count the number of times a hash has co-occurred with each of the t_i in T. For the summation-based method we can consider the number of co-occurrences of hashtag h and the terms in a tweet T.

$$sum(h, T) = \sum_{t_i \in T} count(h, t_i)$$

where $count(h, t_i)$ denotes the number of co-occurrences of hashtag h and the terms in a tweet T. In both methods the scores are additionally normalized with a promotion score in which the stability of the term, descriptiveness of hashtag and the rank of hashtag in the co-occurrence list of the terms is considered. For more explanation we refer the interested reader to [20].

4.2.2 Bayesian Prediction

The second approach is based on Bayesian principles and also originates in image-based tag prediction [27]. Adapting this method, we can consider the co-occurrence of hashtags and terms along with the user tag history. Here, the probability of suggesting a hashtag h to a user u for the resource t_i is defined as:

$$p(h|u, t_i) = \frac{p(u, t_i|h) * p(h)}{p(u, t_i)} = \frac{p(u|h) * p(t_i|h) * p(h)}{p(u, t_i)} \quad (8)$$

where $p(h|u, t_i)$ is the probability that user u uses hashtag h to annotate resource t_i, $p(u, t_i|h)$ is the posterior probability of user u and resource t_i given a hashtag h, and $p(h)$ is the prior probability of hashtag h. Having the score of a hashtag h for each of the terms t_i we can find the total score of a hashtag for the whole tweet T as:

$$p(h|u, T) = \sum_{t_i \in T} p(h|u, t_i) \quad (9)$$

In this method since the user tagging history is taken into account, the score measured for each hashtag is a personalized score.

4.2.3 Association Rule Mining

The third approach is based on market-basket data mining principles for predicting tags [8][26]. In the market-basket model we have a large set of items and a large set of baskets containing a subset of items [1]. We are interested to identify the items that are purchased together frequently in a basket. This model generates the association rule of the form $\{I_1, I_2, .., I_n\} \Rightarrow \{h\}$ meaning that finding $I = \{I_1, I_2, .., I_n\}$ in a basket, there is a good chance of finding h in it. In particular, the popular association rule mining approach can be used to identify interesting relationships among terms and hashtags based on the probability of occurrence of the terms with their related hashtags. Adapting the market-basket model to the hashtag prediction problem, the baskets are the tweets, and the items are the terms and hashtags appearing in a tweet. The goal is to find the most probable hashtags when a set of terms I has been observed in a tweet. In this model we care about the term-hashtag pairs that appear frequently together and are considered to have high *support*. Another metric called $confidence$ implies the probability of finding h knowing that I has occurred. The rules with high confidence and support construct the useful association rules. Here we define $supp(I)$ as the number of tweets in which I has appeared and $conf(I \Rightarrow h)$ as the

probability of using hashtag h when I is observed in a tweet as a set of terms:

$$conf(I \Rightarrow h) = P(h|I) = \frac{supp(I,h)}{supp(I)} \quad (10)$$

which is the number of times the terms I and hashtag h appear together divided by the number of times that the terms I appeared in the training dataset. In this way, association rules are used to find interesting term-hashtag relationships. The length of association rule can vary. In practice, the most interesting rules have a length of less than 3 for short text dataset. Hence, we first extract all possible association rules from the training set, keep only those of length 3 or less. To predict the hashtags for a new tweet, the rules with the same input terms and high confidence and support are used. As an example for the tweet "Freedom for journalism in Iran", we can consider the rules with support more than a threshold (30 in this case), resulting in the following high confidence rules: $\{freedom, iran\} \rightarrow \#iran$, $\{iran, journal\} \rightarrow \#iranelection$. Therefore the suggested hashtags will be $\{\#iran, \#iranelection\}$.

4.3 Experimental Results

We now evaluate the performance of the proposed graph-based approach to predict annotations. To do this we use the metrics described in Section 3.1 and the Twitter dataset described in Section 4.1. In particular, we perform three set of experiments: (i) to estimate the parameters and pivot selection methods used in the graph based approach; (ii) to compare the performance of our approach with the alternate approaches described earlier in this section; and (iii) to analyze the graph-based prediction approach.

4.3.1 Parameter Estimation

We estimate three parameters used by the graph-based approach: (i) the number of hops to consider from a pivot term; (ii) the decay factor (β) for penalizing nodes far from the pivot term; and (iii) the length of the sliding window (Δ). In addition to these parameters, we also compare the two pivot term selection methods – by entropy and by inverse document frequency.

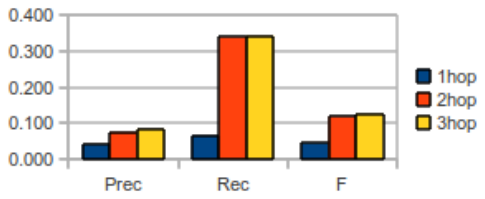


Figure 7: Increasing the number of hops identifies more relevant hashtags in the semantic graph.

Number of Hops: In this experiment we estimate the maximum number of hops to take from a pivot term to determine candidate hashtags for annotation. For example, while a value of one hop will consider only the immediate neighbors of a term, a choice of hops greater than 1 will consider hashtags that do not directly co-occur with the pivot term but are related to it. Hence in this experiment, we tried different number of hops after setting $\beta = 0.80$ and $\Delta = 1$ week. The result of this experiment is shown in Figure 7. We observe a large improvement in recall as the number of hops increases to 2, suggesting that these nearby hashtags are good candidates (even if they have not co-occurred directly with the terms in a particular status update). We also note that the recall and F-measure are nearly the same comparing hops 2 and 3, meaning that additional exploration of the semantic graph identifies few additional significant hashtags. Since this larger exploration comes at a larger computational cost, we set the number of hops to 2 for the remainder of the experiments.

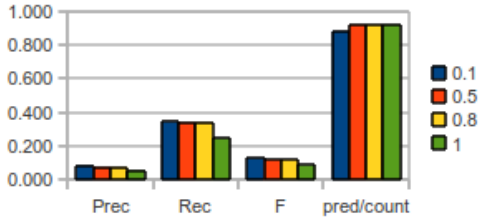


Figure 8: A smaller decay factor results in better performance but fewer overall predictions.

Decay Factor (β): To determine the choice of decay factor β, we set β to values ranging 0.0 to 1.0 and observe the performance of the approach. In addition to β, we set the number of hops to 2 and $\Delta = 1$ week. The result of this experiment is shown in Figure 8. Here we also show the rate of $pc = \frac{pred}{count}$ which measures the proportion of times that the semantic graph-based approach could predict at least one hashtag for the tweets in the test set. We see that a smaller decay factor results in a better performance but fewer overall predictions. Hence, to balance these two factors, we set $\beta = 0.8$.

Δ T	Precision	Recall	F-measure
hourly	0.041	0.042	0.041
hourly4	0.038	0.040	0.039
daily	0.109	0.107	0.107
weekly	**0.174**	**0.261**	**0.203**
monthly	0.132	0.201	0.152

Table 3: Comparing AR predictions with different ΔT. The weekly sliding window builds a better prediction model.

Length of Sliding Window (Δ): We additionally repeated the experiments by varying the length Δ to different values. We set the number of hops to 2 and parameter $\beta = 0.8$. The result of this experiment is shown in Table 3. We see 1 week of sliding window gives the best performance. In comparison, we see that the hourly, 4 hours and daily windows are sparse resulting in poor performance, while a month data tends to recommend outdated hashtags which also results in poor performance.

Approach to Select Pivot Terms: As described in Section 3.2.2, an important problem in the graph-based prediction framework is to select correct pivot terms. We now evaluate the performance of our approach using the two methods – entropy and document frequency. The results are shown in

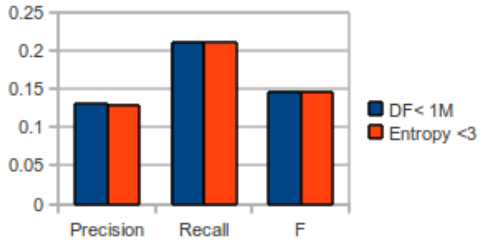


Figure 9: DF and Entropy pivot selection perform nearly equally well.

Method	Precision	Recall	F-measure
promo-vote	0.008	0.025	0.011
promo-sum	0.004	0.014	0.006
bayes	0.023	0.039	0.027
assoc-rule	**0.167**	**0.218**	**0.189**

Table 4: Comparing alternative approaches over 1000 test tweets.

Figure 9. Interestingly, we observe little difference between the performance of these two approaches. Since document frequency is simpler to maintain for all terms, we select it for the remainder of the experiments.

4.3.2 *Comparison of Annotation Prediction Methods*

We next evaluate the effectiveness of the several alternative methods for predicting hashtags. We then present the results of comparing our graph-based approach in detail against the best of these alternative methods.

Comparison of Alternate Methods: The comparison between annotating approaches in [20], [27], and [8], described earlier in this section, is shown in Table 4. For association rules, we report results for $conf = 0.1$ and $sup = 30$; we additionally varied the support threshold between 10 and 100 but found little change in results. We see that the association rule approach results in the best precision, recall, and F-measure (it also is relatively more efficient than the alternate approaches). Intuitively, the association rule approach is effective at weeding out large numbers of weak term-hashtag pairs (via the confidence and support thresholds), resulting in the best relative performance.

Graph-based vs Association Rule: Since association rule mining approach performs the best among the alternate approaches, we now compare it with our graph-based proposed method. Figure 10 compares the association rule based model and the graph-based approach for windows of different lengths. We observe that the association rule approach gives good performance when the length of the sliding windows is large (since it has access to a larger training set to identify term-hashtag relationships). However the graph-based model has a higher recall in all cases and better precision for the shorter sliding windows. These results suggest that the graph-based approach can identify implicit relationships among terms and hashtags by linking terms and hashtags that may have never occurred together.

Combining AR and Graph-Based Approaches: A possible extension of the association rule based model is to combine it with the graph-based annotation prediction method. In this way we could take advantage of the properties of the graph-based model for revealing implicit relationships. Hence, we augmented the term-hashtag association rules discovered by association rule mining by additionally scoring related hashtags using the graph-based approach. In this way, additional hashtags may be identified, offering the possibility of increased recall. We evaluated the performance of this extended version and report the results in Figure 11. While we do observe that the recall of the combined approach is higher than the baseline association rule approach, it is still less than the pure graph-based approach. And disappointingly, the precision of the combined approach is worse than either alternative, suggesting the need for careful future study of the combination of these two approaches.

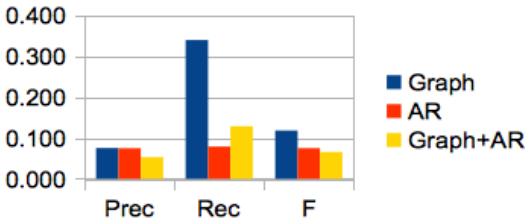


Figure 11: Combining AR with the semantic graph improves recall but not precision.

4.3.3 *Analysis of Graph-Based Approach*

Finally, we turn our attention to analyzing several properties of the graph-based prediction approach and describe a technique to extend its performance using tweet and hashtag categorization.

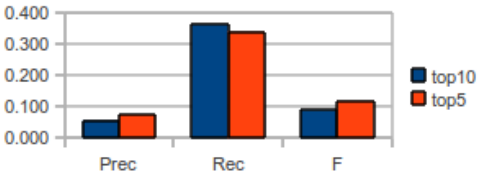
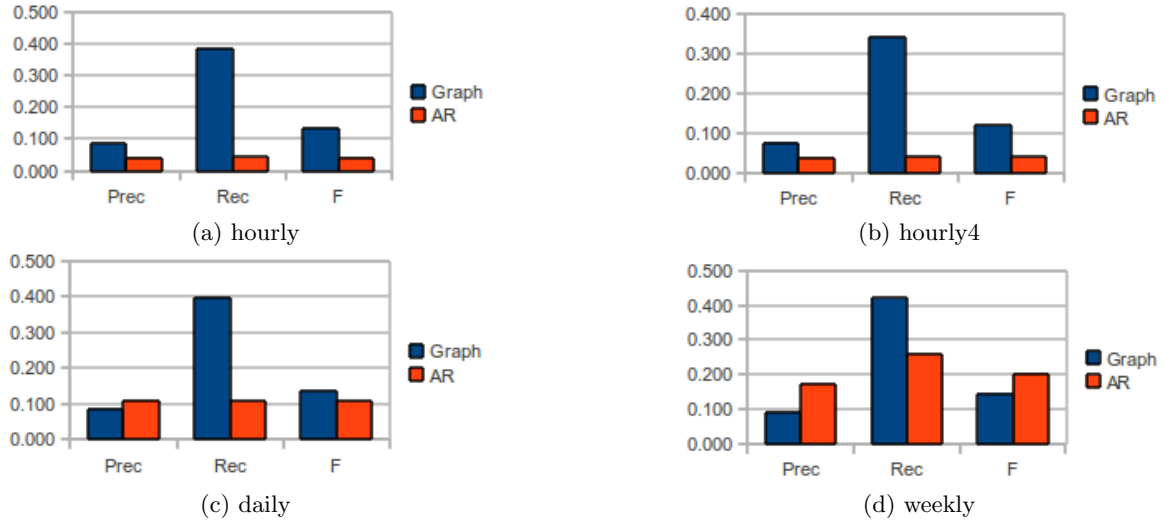


Figure 12: Increasing the number of selected hashtags (topK) lowers precision and increases recall.

Impact of Number of Hashtags: Based on the scores for hashtags generated by our system we select the first *top-K* hashtags. We observe that when *top-K* is small, we have higher precision and when it is larger we have higher recall. We consider *top-K* $= 5$ for the experiments since it gives us a good balance of precision and recall.

Impact of Smoothing: In Section 3, we described a smoothing model considering a mixture of both the most recent window and the global history in terms of hashtag-term linkages in the semantic graph. Performance of this smoothed model with others is shown in Figure 13. We observe a small increase in precision, but almost no improvement in



(a) hourly

(b) hourly4

(c) daily

(d) weekly

Figure 10: Comparing the graph-based and Association Rule based models for different sliding windows. The graph-based approach achieves high recall in all cases and better precision for the shorter sliding windows. The AR approach works well over the longest time horizon, when the training set is the largest.

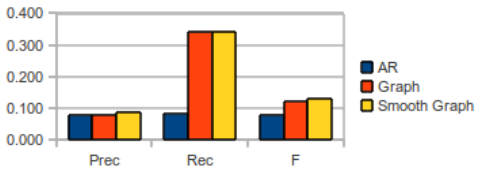


Figure 13: Smoothing increases precision by incorporating longer-term term-hashtag relationships.

recall. Additionally, since we are dealing with more data in the smoothed approach, the time taken to build model is greater, which may be infeasible for real-time annotation as status updates are inserted into the system.

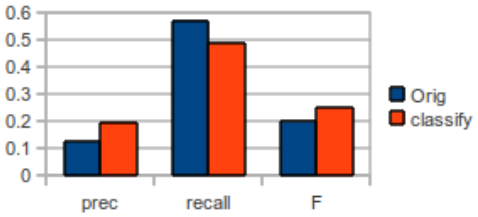


Figure 14: Classification of tweets increases the performance of the baseline approach.

Extending the Approach with Categorical Information: So far, we have studied semantic annotation of status updates using only the content of the updates themselves, without access to additional meta-information about the updates. It may be reasonable to expect that incorporating the category of the update into the prediction framework could increase its performance. Hence, we explore the possibility of improving the predictor by filtering out all suggested hashtags that belong to categories other than the category of the status update itself. Towards this goal, we assume there exists a tweet classifier similar to what is proposed in [19, 23] that can categorize both tweets and hashtags. Here we use the top-500 frequent hashtags that are already labeled by [19] into 8 categories: *Celebrities, Game, Political, Idioms, Music, Movies, Sports, Technology.* Then we consider only the tweets that contain at least one of these labeled hashtags (resulting in 12 million tweets in the dataset). Figure 14 compares this categorical extension with the baseline graph-based model. As expected, we see an increase in precision for the categorical extension, but a decrease in recall. This suggests the potential for incorporating more refined categorical (and perhaps sentiment-based) information into the hashtag prediction framework.

5. CONCLUSION

In this paper, we proposed a graph-based prediction framework for increasing the coverage of semantic annotations in real-time web status updates. We saw how the path aggregation technique for scoring the closeness of terms and hashtags in the graph, pivot term selection, and the dynamic sliding window led to encouraging results in comparison with alternative methods. As systems like Twitter and Facebook continue to grow, the proposed approach could be used to extend the small fraction of self-curated messages to organize the vast majority of messages that have not been annotated. In this way, the feedback between small-scale curation and automated methods may provide an evolving framework for ongoing organization of real-time web content. In our future work, we are interested to augment the baseline model presented here with information from each user's social network, so that hashtags adopted by a user's community may provide a more personalized set of hashtag recommendations. We are also interested to study the impact of increasing spam and low-quality hashtags on the performance of hashtag prediction.

6. REFERENCES

[1] R. Agrawal, T. Imieliński, and A. Swami. Mining association rules between sets of items in large databases. *SIGMOD Rec.*, 22:207–216, June 1993.

[2] D. M. Best, S. Bohn, D. Love, A. Wynne, and W. A. Pike. Real-time visualization of network behaviors for situational awareness. In *Proceedings of the Seventh International Symposium on Visualization for Cyber Security*, VizSec '10, 2010.

[3] A. Bifet and E. Frank. Sentiment knowledge discovery in twitter streaming data. In *Proceedings of the 13th international conference on Discovery science*, DS'10, pages 1–15, Berlin, Heidelberg, 2010. Springer-Verlag.

[4] A. Byde, H. Wan, and S. Cayzer. Personalized tag recommendations via tagging and content-based similarity metrics. In *Proceedings of the International Conference on Weblogs and Social Media*, March 2007.

[5] P. A. Chirita, S. Costache, S. Handschuh, and W. Nejdl. PTAG: Large Scale Automatic Generation of Personalized Annotation TAGs for the Web. May 2007.

[6] N. Garg and I. Weber. Personalized, interactive tag recommendation for flickr. In *Proceedings of the 2008 ACM conference on Recommender systems*, RecSys '08, 2008.

[7] S. Golder and B. A. Huberman. The structure of collaborative tagging systems. *Journal of Information Science*, 32:198–208, 2006.

[8] P. Heymann, D. Ramage, and H. Garcia-Molina. Social tag prediction. In *SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, 2008.

[9] J. Illig, A. Hotho, R. Jäschke, and G. Stumme. A comparison of content-based tag recommendations in folksonomy systems. In *Proceedings of the First international conference on Knowledge processing and data analysis*, KONT'07/KPP'07.

[10] D. Irani, S. Webb, C. Pu, and K. Li. Study of Trend-Stuffing on Twitter through Text Classification. 2010.

[11] R. Jäschke, L. Marinho, A. Hotho, S.-T. Lars, and S. Gerd. Tag recommendations in social bookmarking systems. *AI Commun.*, 21, 2008.

[12] K. Y. Kamath and J. Caverlee. Transient crowd discovery on the real-time social web. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pages 585–594, New York, NY, USA, 2011. ACM.

[13] G. Koutrika, F. A. Effendi, Z. Gyöngyi, P. Heymann, and H. Garcia-Molina. Combating spam in tagging systems: An evaluation. *ACM Trans. Web.*

[14] K. Lee, J. Caverlee, K. Y. Kamath, and Z. Cheng. Detecting collective attention spam. In *Proceedings of the 2nd Joint WICOW/AIRWeb Workshop on Web Quality*, WebQuality '12, 2012.

[15] P. Mika. Ontologies are us: A unified model of social networks and semantics. Lecture Notes in Computer Science, pages 522–536. Springer, 2005.

[16] G. Mishne. In *Proceedings of the 15th international conference on World Wide Web*, New York, NY, USA.

[17] M. Naaman, J. Boase, and C.-H. Lai. Is it really about me?: message content in social awareness streams. In *Proceedings of the 2010 ACM conference on Computer supported cooperative work*, CSCW '10, 2010.

[18] A. Rae, B. Sigurbjörnsson, and R. van Zwol. Improving tag recommendation using social networks. In *Adaptivity, Personalization and Fusion of Heterogeneous Information*, RIAO '10, 2010.

[19] D. M. Romero, B. Meeder, and J. Kleinberg. Differences in the mechanics of information diffusion across topics: idioms, political hashtags, and complex contagion on twitter. WWW '11, 2011.

[20] B. Sigurbjörnsson and R. van Zwol. Flickr tag recommendation based on collective knowledge. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, 2008.

[21] Y. Song, Z. Zhuang, H. Li, Q. Zhao, J. Li, W.-C. Lee, and C. L. Giles. Real-time automatic tag recommendation. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, 2008.

[22] S. C. Sood and K. J. Hammond. Tagassist: Automatic tag suggestion for blog posts. In *In International Conference on Weblogs and Social*, 2007.

[23] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, and M. Demirbas. Short text classification in twitter to improve information filtering. SIGIR '10.

[24] J. Sun, C. Faloutsos, S. Papadimitriou, and P. S. Yu. Graphscope: parameter-free mining of large time-evolving graphs. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2007.

[25] J. Teevan, D. Ramage, and M. R. Morris. #twittersearch: a comparison of microblog search and web search. WSDM '11, 2011.

[26] J. Wang, L. Hong, and B. D. Davison. Rsdc'09: Tag recommendation using keywords and association rules.

[27] Z. Wang and Z. Deng. Tag recommendation based on bayesian principle. In *Proceedings of the 6th international conference on Advanced data mining and applications*.

[28] Z. Xu, Y. Fu, J. Mao, and D. Su. Towards the semantic web: Collaborative tag suggestions. In *Proceedings of Collaborative Web Tagging Workshop at 15th International World Wide Web Conference*, 2006.

[29] J. Yang and J. Leskovec. Patterns of temporal variation in online media. In *WSDM*, pages 177–186, 2011.

[30] Z.-K. Zhang, T. Zhou, and Y.-C. Zhang. Personalized recommendation via integrated diffusion on user-item-tag tripartite graphs. 2009.