Predicting Top-k Trends on Twitter using Graphlets and Time Features

Gustav Šourek, Ondřej Kuželka, and Filip Železný

Faculty of Electrical Engineering, Czech Technical University in Prague
Technicka 2, 16627 Prague, Czech Republic
{souregus,kuzelon2,zelezny}@fel.cvut.cz,

Abstract. We introduce a novel method for predicting trending keywords on Twitter. This new method exploits topology of the studied parts of the social network. It is based on a combination of graphlet spectra and so-called time features. We show experimentally that using graphlets and time features is beneficial for the accuracy of prediction.

1 Introduction

In this paper, we present a method which exploits information about graph structure of sub-networks of the social network Twitter for making better predictions about which topics will get among the top k. We show experimentally that with information about the graph structure we are able to obtain better predictive accuracies than with a model trained on the same data which does not use information about the graph structure. Importantly, the presented method does not need access to the entire graph structure as it works only with certain sets of derived attributes, which makes it potentially possible to combine the method with sampling strategies and also to make it able to work in differentially private settings.

As the network structure has been proved to play an important role in spreading social trends [1], we want to exploit the effect of social network topology beyond the scope of previous works (e.g. beyond merely measuring nodes' degrees, centrality etc.). Inspired by creating network signatures from graphlet degree distributions [2] in biological networks, we use similar representation to reflect a trend presence within our network. For that we create graphlet features - small connected subgraphs, representing various local relative topology options, and measure their presence in the network by means of subgraph matching. The network trend signatures, calculated from the frequencies of respective features occurrences, are then used as feature vectors for standard machine learning algorithms.

2 Problem Setting

Twitter is an online social networking service that enables its users to send and read text-based messages of up to 140 characters known as *tweets*. At the same

time it enables users to connect to others through the *follows* relationship. The users that a particular user is following through this relation are referred as his friends. Users on the other side who are following the particular user are referred simply as his followers. Tweets posted by a particular user are stored and displayed as a chronological sequence in user's timeline. Each such a tweet being posted is also broadcasted to the users followers. Tweets are, by default, public, which means that anyone can list them out through Twitter's search engine or other Twitter API facilities and join the related conversation. Moreover Twitter users can engage in a direct conversation between each other. As for the information content, users can group posts together by type with the use of *hashtags* - words or phrases prefixed with a "#" sign, referring a tweet to the specified topic. Hashtag signed tweets have special treatment in Twitter's engine and can be easily searched out.

Now, we define the prediction problem that we will be dealing with in this paper, namely the problem of predicting the top-k trends. Unlike original Twitter engine, we consider trending topics clearly by measuring the frequency of occurrence of corresponding hashtag in the network. If the relative frequency of hashtag in a particular timeframe is among the top-k, we declare it a trend. One can imagine a web page which gives its users a list of k hashtags which are predicted to get among the hottest topics in his subnetwork in the near future.

Learning is performed on data as a time series, where each hashtag occurrence information goes into a prepared *time-fold* according to its time of creation. The task is to predict which hashtags will be trending in the future target time-folds (determined to day intervals). Unlike in the case of the basic supervised learning task, there is an additional constraint on the output of the classifier. On every single day, the classifier must mark as trending exactly k hashtags. To satisfy this constraint, the classifier takes the probability distribution of classification given by the learned Random Forest [3] model and creates respective ranking on the instances that are subject to the current prediction. That means that only the top k instances classified with highest confidence as trending will be considered positive. This k-set will then be compared with the true top-k list for the target day. The natural measure of quality of such a prediction, denoted as top-k% metric here, is the percentage of correctly predicted topics in the target list.

3 Simple and Baseline Models

In order to assess the contribution of graphlets and time features to accuracy of prediction, we created two models which we call *simple model* and *baseline model* and which serve as a baseline in this paper. In these approaches we wittingly ignore the social graph structure and take the problem as a time series prediction, which is the case of most methods found in literature. The simple model method represents the common sense approach to the statistics measured. It builds on basic average occurrence of the hashtags, calculated over all time-folds in the training part of the time-window, and treats them as if they were to

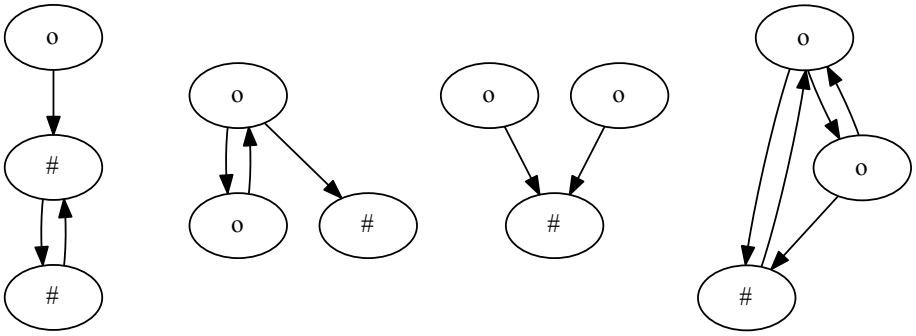

Fig. 1. Selected examples of features of size = 3, with some of the highest information gains with respect to the trends spread.

continue constantly with that occurrence in the target part. The baseline learner represents a classical time-series prediction method, with the use of a model to predict future values based on previously observed values in data with natural temporal ordering. To make a clear comparison of the models used, we turn the time series forecasting task into a standard classification problem, using a sliding window technique [4]. Since our main focus lies with the features extracted, it always uses the same machine learning algorithm as the graph learner.

4 A Model Based on Graphlets and Time Features

The term *graphlet*, as used in this paper, refers to a small directed graph (with up to 3 nodes) which contains at least one node labelled by "#". Examples of graphlets are shown in Figure 1. For every day D and every hashtag H, the *snapshot* of the sub-network is a directed labelled graph constructed as follows. There is one node for every user. There is a directed edge between two users U_i and U_j if and only if the user U_i *follows* the user U_j. A node is labelled by "#" if and only if the user corresponding to this node used the hashtag H on the day D in at least one tweet. Given a list of graphlets L_g and a snapshot S of the sub-network we can construct a so-called *frequency-feature vector* as follows. For every graphlet $g_i \in L_g$, we count the number of homomorphisms of g_i to the snapshot graph S (respecting the labels "#") and store the number in the i-th element of the frequency-feature vector. Clearly, frequency-feature vectors are not very suitable for prediction of top-k trends because their values are sensitive to the overall activity of the users on the given day. Therefore we need so-called *rank-feature vectors* which can be constructed from the frequency-feature vectors. Given a set of frequency-feature vectors for all hashtags of interest on a given day, the i-th element of a rank-feature vector V_{rank} for a hashtag H is the rank (i.e. order) of the respective i-th element of the frequency-feature vector V_{freq} corresponding to the same H among all i-th elements of the other frequency-feature vectors corresponding to the same hashtag H. Given a time

window consisting of several days, one can easily create a graphlet representation by concatenating the rank-feature vectors corresponding to these days.

The rationale behind the graphlet model is that graphlets can capture how natural user to user connections in Twitters network affect the topics discussed. They consist of nodes and edges representing occurrence of hashtag on users time-line in context of his neighbors, e.g. his followers and friends. A somewhat similar representation was used in the work of Nataša Pržulj for computing a network structure similarity measure using so called graphlet degree distribution [2], where graphlets were small connected non-isomorphic induced subgraphs of a large network. Our relational approach differs in that, with our features, we generate the subgraphs separately, in advance of further matching in the whole network, while we do not restrict them to be induced. Even more importantly, as far as we know, our approach is the first to use graphlets to model dynamical processes in complex networks.

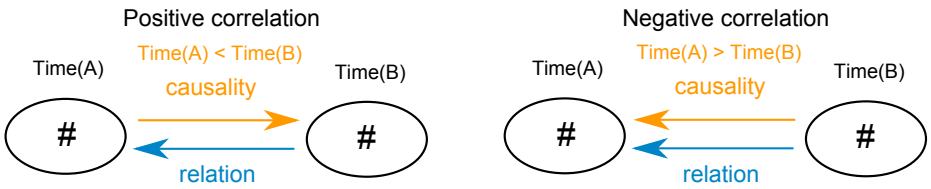

Fig. 2. The two cases of correspondence between the relation and causality in hashtag spreading.

Besides graphlets we use also so-called *time features*. The main purpose of time features is to add a measure of some time properties of the underlying networks relations. The motivation for this comes from a natural intuition of trends spread in social networks. In a directed network like Twitter, if the information is being spread through the network, the fashion of the spreading should correspond to the network structure. By that we mean that the directed relations between the users should actually represent the causality links in the trend spread dynamics. If it is not the case, the information is probably not coming from the network and is being spread by other channels. Now how to measure this networks causality correspondence? For a snapshot corresponding to a hashtag H, we label all the directed edges $e = (U_i, U_j)$ which connect two nodes U_i and U_j both of which are labelled by "#", by the time between the instant when U_i posted the tweet with the hashtag H and the instant when U_j posted a tweet with the same hashtag. Since the difference is measured against the direction of the underlying relation, it can in general be negative as depicted in Figure 2. The time features can then be computed as averages and standard deviations of these time differences by which the edges are labelled.

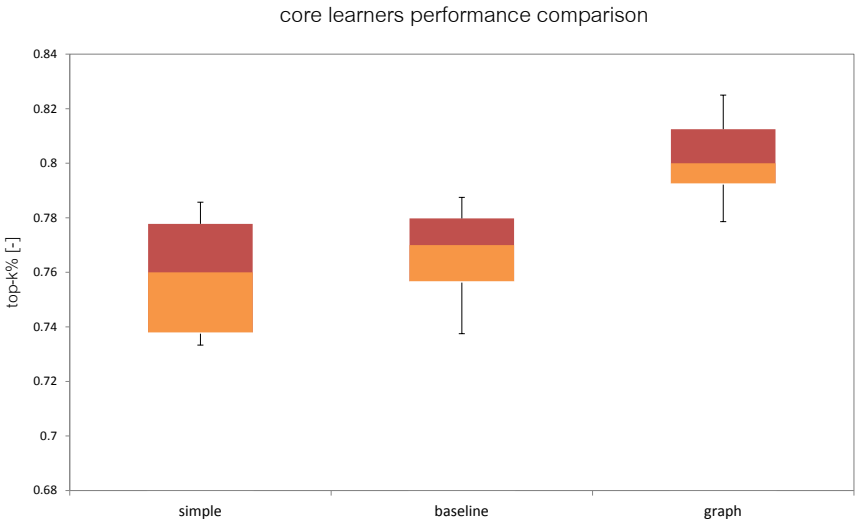

Fig. 3. Final comparison of core learners on the top-k% measure.

5 Experimental Evaluation

For the testing of our graph-based approach we needed to download a suitable dataset of tweets from Twitter. We implemented a simple crawler using the Twitter API. The sampling strategy, i.e. the choice of the set of users that should be downloaded was driven by a simple heuristic. Starting from a random seed, the heuristic orders the nodes (corresponding to users) whose neighbours should be added to the database in a greedy way so that the crawled sub-network would be as compact as possible – it picks preferentially those nodes which share most edges with nodes already stored in the database. The network subsets that were eventually crawled consist of approx. 8 thousand users with 3 million internal connections, and millions of public tweet (hashtag containing) records from December 2012, March and April 2013.

We measured the performance of our novel method and the simple and baseline methods on the largest dataset from March 2013, with windows consisting of 4 days of training and 1 day for prediction. We set k to 20 and performed multiple runs of the classifier with varying seed. We tested our features with two classifiers, namely SVM and Random Forest, both giving similar results, yet the later proved more suitable for tuning and time complexity reasons. The choice of parameters was tuned as to avoid overfitting of the classifier, i.e. by extending the training timescope to at least 4 days, and to have a clear threshold to cut between trending and non-trending hashtags, i.e. higher k such as 20 helped to avoid the situation of constant classification confidence for all the trending hashtags.

The improvement on the top-k% metric achieved by our method, as displayed in Figure 3, might in reality correspond to early detection of a couple more upcoming trends that wouldn't be otherwise discovered without taking the structure of the network into account.

We also performed other experiments with the novel method which we do not report in detail here due to lack of space but which can be found in [5]. For example we evaluated the approach on different metrics, trend definitions and parameters. We examined the influence of various timescope settings, e.g. the size of train and test parts of the sliding window and various time-fold granularities. We tested the resilience of the approaches to change of data content and network structure by an interchange of training sets from multiple datasets, proving reasonable sensitivity of the graphlet approach both to the change of the content (negligible) and to the change of the structure (slightly bigger sensitivity). We also tested whether we could not obtain better results with other relations, but the results came in the favour of the original *follows* relation over the *retweets* and *replies*. We also assessed the usefulness of time features. It turned out that time features contribute to the performance in the order of several percent. Nevertheless, time features on their own performed worse than graphlets.

6 Conclusions

In this paper, we presented an approach for prediction of trends spread within a local Twitter subnetwork, utilizing topology structure information, based on representation, inspired by methods from the area of biological networks. The results prove the value of knowledge on the network structure and the contribution of the approach itself. One of the appealing properties of the method is that it exploits information about structure of the network but at the same time it does not need the entire structure nor it does need to construct models of individual users.

Acknowledgements

This work was supported by the Czech Science Foundation grant no. P202/12/2032 and the Czech Technical University internal grant SGS11/155/OHK3/3T/13.

References

1. NA, C., JH, F.: Social network sensors for early detection of contagious outbreaks (2010) PLoS ONE 5(9): e12948.
2. Tijana Milenkovic, N.P.: Uncovering biological network function via graphlet degree signatures (2006) Oxford university press.
3. Breiman, L.: Random forests (2001) Statistics Department, University of California, Berkeley, CA 94720.
4. Dietterich, T.G.: Machine learning for sequential data: A review (2010)
5. Šourek, G.: Twitters local trends spread analysis (2013) `http://cyber.felk.cvut.cz/research/theses/detail.phtml?id=339`.