Filtering Relevant Facebook Status Updates for Users of Mobile Devices

Stephan Baumann[1], Rafael Schirru[1,2] and Joachim Folz[2]
[1]*German Research Center for Artificial Intelligence, Trippstadter Straße 122, 67663 Kaiserslautern, Germany*
[2]*University of Kaiserslautern, Gottlieb-Daimler-Straße 47, 67663 Kaiserslautern, Germany*

Keywords: Content-based Information Filtering, Social Networks, Mobile Devices, Classifier-based Approach.

Abstract: In recent years, social networking sites such as Twitter, Facebook, and Google+ have become popular. Many people are already used to accessing their individual news feeds ubiquitously also on mobile devices. However the number of status updates in these feeds is usually high thus making the identification of relevant updates a tedious task. In this paper we present an approach to identify the relevant status updates in a user's Facebook news feed. The algorithm combines simple features based on the interactions with status updates together with more sophisticated metrics from the field of Social Network Analysis as input for a Support Vector Machine. Optionally the feature space can be extended by a topic model in order to improve the classification accuracy. A first evaluation conducted as live user experiment suggests that the approach can lead to satisfying results for a large number of users.

1 INTRODUCTION

Online social networks such as Facebook, Twitter, and Google+ have experienced an enormous growth in their user bases in the last years. They are ubiquitous and people have become used to accessing them on a variety of different devices such as PCs, tablets, and smartphones. With an average of 130 friends per user[1] and the introduction of frictionless sharing many users of the Facebook platform experience an information overload, making the manual identification of relevant status updates in their news stream a tedious task. Accessing the platform from mobile devices with limited screen size aggravates the problem.

Facebook does advanced ranking for personalized placement of objects in the news feed of a user using a proprietary and trademarked algorithm. The EdgeRank TM is based on the affinity of a user to objects and subjects in the Facebook system as well as the weight of objects and relevance over time, i.e. decay. Several U.S. patents describe rather vaguely how the algorithm combines affinity, weight and decay. The critics of this approach claim that Facebook is using EdgeRank TM in order to control the flow of information to optimize their own business model. Facebook does not offer API access for third-party developers or researchers.

Therefore in this paper we present our approach to identify relevant status updates in a user's Facebook news feed including algorithmic details. The method combines simple features based on the interactions with status updates together with more sophisticated metrics from the field of Social Network Analysis as input for a Support Vector Machine. Optionally a topic model can be added to the feature space.

2 RELATED WORK

In order to determine the status updates that are relevant for a particular user, we consider the extraction of opinion leaders (friends with a high authority in the user's network) as well as the identification of the flow of information (content that is shared in the user's social network) as important aspects. In order to measure the importance or authoritative power of linked objects in a network the well-known PageRank (Page et al., 1998) and Kleinberg's HITS algorithm (Kleinberg, 1998) are de-facto standards. While PageRank is using the Random-Surfer model to compensate real-world effects in large networks, HITS is well-suited for computations of its hub and authority values in smaller networks. For this reason we chose HITS over PageRank for the implementation of relevance in our approach. Further we find the following work particularly relevant for our method:

(Roch, 2005) identified two different sets of at-

[1] http://www.facebook.com/press/info.php?statistics

Baumann S., Schirru R. and Folz J..
Filtering Relevant Facebook Status Updates for Users of Mobile Devices.
DOI: 10.5220/0004188003530356
In *Proceedings of the 5th International Conference on Agents and Artificial Intelligence* (ICAART-2013), pages 353-356
ISBN: 978-989-8565-38-9

tributes that characterize opinion leaders. The attributes in the first set are specific for the person (e.g., the sources of information he/she relies on). Attributes from the second set characterize the milieu of a person. Roch's research suggests that opinion leaders have an information advantage relative to the others in their environment.

(Boccara, 2008) investigates the formation of opinions in populations under the influence of opinion leaders. An agent-based model is used to simulate the adoption of opinions under several adoption rules over time. An agent's opinion is influenced by its neighbors and its awareness, where a low awareness means an individual is very likely to adopt new opinions and vice versa. Opinion leaders are then selected from those individuals with a maximum number of individuals they influence and with equal probability given one of two distinct opinions. They also possess maximum awareness and thus never change their opinion.

(Weng et al., 2010) present TwitterRank, an algorithm identifying influential Twitter users. The algorithm works as follows: First it extracts the topics of tweets based on their content. In the second step topic-specific relationship networks are constructed among the twitterers. Finally, the TwitterRank algorithm is applied which is an adaptation of the PageRank algorithm taking the topical similarity between twitterers and the link structure into account.

3 APPROACH

The approach presented in this section aims at identifying the relevant status updates in a user's Facebook news feed. We set the terminology as follows: The *user* is the person whose news feed is currently being analyzed. The people in the user's network are referred to as *friends*. Expressing a "like" for a status update or commenting on a status are referred to as *interactions*.

3.1 Crawling the User's News Feed

In the first step we crawl the status updates and their associated interactions (comments and likes) from the user's Facebook news feed and store it in a relational database. We use the Graph API[2] in order to obtain the required data. Currently our system supports status updates of the following types: message, photo, video, link, new friendship, and check-in.

3.2 Feature Extraction

From the user's news feed we extract features that are associated with a friend, e.g, her hub and authority values. We call these features friend-specific features as they are the same for all status updates of the respective friend. Further features are extracted that are specific for each status update, e.g., the number of interactions a particular status update has. These features are called status-specific features. The features will be described in more detail subsequently.

3.2.1 Interaction Percentage

The interaction percentage is a friend-specific feature. It determines the relative amount of interactions a user had with the status updates of a particular friend. Let S_f be the number of status updates friend f has posted in the user's crawled news feed and let I_f be the number of status updates of friend f the user has interacted with. The interaction percentage F_{ip} is calculated as follows:

$$F_{ip} = \frac{I_f}{S_f} \tag{1}$$

3.2.2 Hub and Authority Values

For each friend we calculate the *hub* and *authority* scores following (Kleinberg, 1998). To determine these values we interpret the user and his/her friends as nodes in a directed network. Each interaction on a status update is interpreted as an edge from the originator of the interaction to the publisher of the status update. Figure 1 shows an example of actions and interactions in a user's personal network. Following Kleinberg an authority in such a network is a node with a high amount of influence, i.e., a node having many incoming edges. In our scenario this corresponds to a large amount of interactions on the person's status updates (cf. friends on the right hand side of Figure 1). A hub on the other hand is considered a node that links to authoritative nodes. In our scenario this means that a hub often interacts with influential persons in the user's network (cf. friends on the left hand side of Figure 1).

3.2.3 First Publisher

The first publisher feature (F_{fp}) is a Boolean flag that indicates whether a friend has contributed one or more status updates as the first person in a user's network that have been shared by another person afterwards. It is intended to measure the information advantage of a Facebook friend (Roch, 2005). $F_{fp} = 1$ if the friend is a first publisher and $F_{fp} = 0$ otherwise. Obviously, the

[2] http://developers.facebook.com/docs/reference/api/

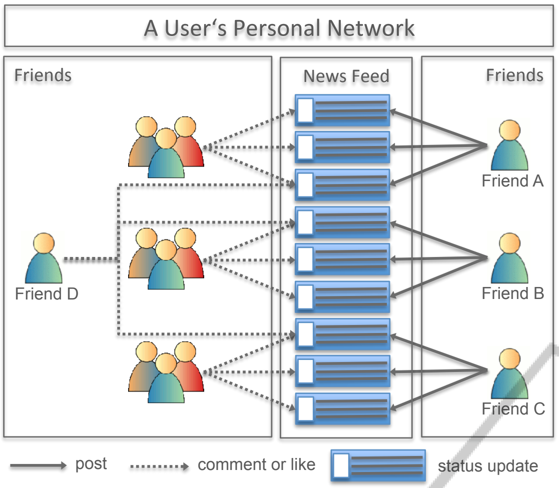

Figure 1: Example of actions and interactions in a user's personal network.

first publisher feature belongs to the friend-specific features.

3.2.4 Interaction Count

This feature measure the number of interactions a status update has received from the people in the user's network. The interaction count is a status-specific feature.

Please note that interactions outside the user's network are not taken into account. For that reason status updates of celebrities or companies the user follows do not necessarily have a high interaction count, when not many of the user's friends follow the respective pages and interact with their published content.

3.2.5 Topic Model

From the status updates in the user's Facebook news feed we calculate a topic model using Latent Dirichlet Allocation (LDA) (Blei et al., 2003). Before applying the algorithm each status update is preprocessed comprising stop word removal and stemming for English and German updates. Further string preprocessing steps are conducted (e.g., terms are converted to lower case characters, punctuation characters are removed). Then the LDA algorithm is run. Each identified topic is represented as a dimension in the feature space with a boolean value indicating whether the status update that is currently being analyzed is associated with the topic or not.

3.3 Training the Classifiers

We use Support Vector Machines (SVMs) (Cortes and Vapnik, 1995) with an RBF kernel in order to classify Facebook status updates as relevant or irrelevant. The software library used for this purpose was the Java implementation of LIBSVM.[3] We trained two different configurations of the SVM using the features described above. The features associated with the topic model were however only used with the second configuration. All features were normalized to values between 0 and 1 before fed to the classifier.

4 EVALUATION

4.1 Data Set

We conducted a live user evaluation experiment with twelve participants that were between 19 and 48 years old. For this purpose users with more than 100 friends in their Facebook network were selected thus making sure that enough status updates could be obtained for each user.

We crawled the last 60 updates in each user's news feed and deleted those that did not have a text message attached. For the experiment we aimed at having 50 status updates for each user. The data was split into 30 instances for the training set and 20 instances for the test set. If less than 50 status updates could be obtained the training set was reduced accordingly. For two users less than 40 updates with a message attached were available. We excluded these users from the experiment as the data was not sufficient for a reliable analysis.

4.2 Procedure and Measures

First, the users had to rate the status updates from their training set. The rating was on a binary scale indicating whether a status update was relevant for a participant or not. For those status updates with additional information (e.g., photos, or videos) we provided links where the users could access the actual content. After the training set was rated, a topic model was created and the classifiers were trained as described in Section 3.3.

When building the topic model we set the number of topics that had to be extracted to four. Each topic was represented by one dimension in the input matrix for the support vector machine. We selected a relatively small topic number in order to avoid an overstated influence of the topic model over the graph based features. We set the document topic prior α to 0.1 and the topic word prior β to 0.01.

[3] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

Then the predictions of the classifiers were calculated for the test set and stored in the data base. In the next step the users rated the status updates in their test set, again on a binary scale. For each classifier we determined the precision, recall and f-measure.

In the next section we compare the results of our method using a feature space without the topic model against the results using a feature space that includes the topic model. It would have been worthwhile to compare these results with Facebooks EdgeRank TM algorithm, however to the best of our knowledge a user's news feed filtered according to the EdgeRank TM algorithm is not available via the Graph API.

4.3 Results

The SVM trained with the feature space including the topic model made better results in terms of our selected measures. Here our approach achieved an average precision of 0.51 at an average recall of 0.47 and an average f-measure of 0.48. It should be noted that for two participants the precision and recall values were 0. When talking to those users after the experiment, it turned out that they did not have many relevant status updates in their training set thus making the learning of a model with a high predictive accuracy difficult. We assume that with more training data the filtering for these participants could be improved.

The results for the SVM trained with the feature space without topic model were slightly worse. In that setting we achieved an average precision of 0.46 at an average recall of 0.38 and an average f-measure of 0.39. Using the reduced feature space three participants had precision and recall values of 0.

For both settings six of the ten participants could achieve an f-measure of 0.5 or better. The results for each participant and the two feature spaces applied are depicted in Table 1.

5 CONCLUSIONS

In this paper we presented an approach to identify the relevant status updates in a user's Facebook news feed. The method combines features based on the interactions with status updates together with metrics from the field of Social Network Analysis, and an LDA topic model as input for a Support Vector Machine. A first evaluation conducted as laboratory study suggests that the approach can lead to satisfying results for a large number of users. However a larger field study is needed to further prove the usefulness of the filtering algorithm.

Table 1: Results of the evaluation experiment (PRC: Precision, RCL: Recall, F: F-Measure). Each row represents the results of one participant. The last row shows the column averages.

W/o Topic Model			***Topic Model***		
PRC	*RCL*	*F*	*PRC*	*RCL*	*F*
0.667	0.4	0.5	0.545	0.6	0.571
1	0.571	0.727	0.75	0.857	0.8
0	0	0	0.833	0.5	0.625
0	0	0	0	0	0
0.714	0.714	0.714	0.722	0.929	0.812
0.529	0.818	0.643	0.5	0.545	0.522
0	0	0	0	0	0
0.455	0.556	0.500	0.429	0.333	0.375
0.5	0.083	0.143	0.5	0.25	0.333
0.733	0.688	0.71	0.786	0.688	0.733
0.46	0.383	0.394	**0.507**	**0.47**	**0.477**

ACKNOWLEDGEMENTS

This research has been funded by the Investitionsbank Berlin in the project "Voice2Social", and co-financed by the European Regional Development Fund.

REFERENCES

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Boccara, N. (2008). Models of opinion formation: influence of opinion leaders. *Int. J. Mod. Phys. C*, 19(1):93–109.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Mach. Learn.*, 20(3):273–297.

Kleinberg, J. M. (1998). Authoritative Sources in a Hyperlinked Environment. In *Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 668–677. AAAI Press.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1998). The pagerank citation ranking: Bringing order to the web. Technical report, Stanford University.

Roch, C. H. (2005). The dual roots of opinion leadership. *Journal of Politics*, 67(1):110–131.

Weng, J., Lim, E.-P., Jiang, J., and He, Q. (2010). Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, pages 261–270, New York, NY, USA. ACM.