TELKOMNIKA, Vol.14, No.3, September 2016, pp. 1052~1058
ISSN: 1693-6930, accredited A by DIKTI, Decree No: 58/DIKTI/Kep/2013
DOI: 10.12928/TELKOMNIKA.v14i3.3150

A Comparison of Retweet Prediction Approaches: The Superiority of Random Forest Learning Method

Hendra Bunyamin*[1], Tomas Tunys[2]
[1]Department of Informatics, Maranatha Christian University
Jl. Prof. drg. Suria Sumantri No. 65 Bandung, Indonesia, Telp/Fax:+62-222012186/222005915
[2]Department of Computer Science, Czech Technical University
Zikova 1903/4 166 36 Prague 6, Czech Republic, Telp: +420-224357576
*Corresponding author, e-mail: hendra.bunyamin@it.maranatha.edu[1], tunystom@fel.cvut.cz[2]

Abstract

We consider the following retweet prediction task: given a tweet, predict whether it will be retweeted. In the past, a wide range of learning methods and features has been proposed for this task. We provide a systematic comparison of the performance of these learning methods and features in terms of prediction accuracy and feature importance. Specifically, from each previously published approach we take the best performing features and group these into two sets: user features and tweet features. In addition, we contrast five learning methods, both linear and non-linear. On top of that, we examine the added value of a previously proposed time-sensitive modeling approach. To the authors' knowledge this is the first attempt to collect best performing features and contrast linear and non-linear learning methods. We perform our comparisons on a single dataset and find that user features such as the number of times a user is listed, number of followers, and average number of tweets published per day most strongly contribute to prediction accuracy across selected learning methods. We also find that a random forest-based learning, which has not been employed in previous studies, achieves the highest performance among the learning methods we consider. We also find that on top of properly tuned learning methods the benefits of time-sensitive modeling are very limited.

Keywords*: retweet prediction, machine learning algorithms, performance*

1. Introduction

Social media like Twitter has provided a platform for spreading information among users [1, 2]. In this work we focus on the retweet prediction problem. Given a tweet, we would like to predict whether it will be retweeted. Applications of this task are, for example, to help decision makers propagate their issues and facilitate companies to promote their products.

A wide range of learning methods and features have been proposed for retweet prediction; see, e.g., [3-7]. In addition, different modeling setups have been proposed; e.g., Petrovic, et al., [3] propose a time-sensitive model that builds separate models depending on the tweet's creation time and show that it substantially improves performance over the passive-aggressive learning algorithm. This large variety of features, learning methods, and ways of modeling the task calls for a systematic comparison on a single dataset. Therefore, we propose a systematic comparison of the selected learning methods and features, both within and without time-sensitive framework, all on a single dataset. To the authors' knowledge this is the first attempt to collect best performing features and contrast linear and non-linear learning methods.

We consider the following research questions: (i) Which of the proposed learning methods is the most effective for the retweet prediction task? (ii) Which of the proposed features are most discriminative features for the learning methods considered? And (iii) To which degree does time-sensitive modeling help improve the performance of learning methods on our dataset?

Prior work has documented several techniques to solve the retweet prediction problem. Naveed, et al., [5] state that the problem of finding "interestingness" on Twitter is the same as predicting whether a tweet will be retweeted. They employ logistic regression to do the prediction and find that content features such as initial negative sentiments make a tweet more likely to be retweeted.

Received November 25, 2015; Revised June 14, 2016; Accepted June 29, 2016

Petrovic, et al., [3] cast the retweet prediction problem as a binary classification problem with a time-sensitive modeling approach. They argue that time-sensitive modeling substantially helps improve performance over passive-aggressive algorithm on their dataset. They also claim that social features that are related to a user improve the accuracy of their model more than tweet features. Our work utilizes the time-sensitive (TS) modeling; however, the TS modeling in out dataset has limited contribution for our prediction task.

Predicting retweets and replies for a given tweet also done by Artzi, et al., [6]. Specifically, for a given tweet, they discover that removing social features such as number of followers, number of followees, and ratio between the two causes a big drop in their model's prediction accuracy over models that include them. Caruana, et al., [8] mention random forest-based learning method without calibration give the best average performance across all metrics and test problems; moreover, Fernandez-Delgado, et al., [9] construct thorough experiments and conclude random forest-based learning method achieve the maximum accuracy. Therefore, we opt to utilize random forest-based learning method rather than employ MART in our work. Moreover, it turns out that random forest-based learning method, which has not been employed in previous studies, gives superior results in our comparison study.

Hong, et al., [4] predict whether or not a tweet will be retweeted and how many times a tweet will be retweeted. They employ logistic regression in their work. How many times a new tweet will be retweeted based on a certain threshold is studied by Jenders, et al., [10]. They utilize models such as Naive-Bayes and generalized linear model. Other work by Gao, et al., [11] also predict how many times a tweet will retweeted by applying an extended reinforced Poisson process model with time mapping process.

Zaman, et al., [12] measures the popularity of a tweet through the time-series path of its retweets. A Bayesian probabilistic model is developed for the evolution of the retweets and popularity of a tweet is predicted based on the retweet times and local network or "graph" structure of retweeters. Macskassy, et al., [13] tag tweets with Wikipedia categories and generate profiles of "topics of interest" based on past content posted and construct retweet behavior models for users. They argue that people's retweeting behavior is better explained through multiple different models rather than one model.

Morchid, et al., [14] study the behavior of tweets that have been *massively* retweeted in a short time. Specifically, they employ Principal Component Analysis to select features. Compared to our work, they extract less number of features and number of learning methods.

Xu, et al., [7] analyze user retweet behavior at individual level and argue that the most important features for general people are social. Our work is similar to theirs; however, our focus is specifically to understand tweets from politicians and we employ more algorithms that have not been tested, specifically random forest-based learning method, and more modeling in our experiments. Moreover, the random forest-based learning method that we employ gives better prediction accuracy than other algorithms they employ.

2. Modeling and Features

In this section we describe our modeling approach for addressing the retweet prediction problem and cast the problem as a binary classification problem.

2.1. Modeling for Retweet Prediction

Table 1 describes 4 groups of learners that we employ in our modeling approach.

Table 1. Learning approaches.

Group of learners	Model
Global linear	Global passive-aggressive (G-PA)
	Global linear support vector (G-LSV)
	Global logistic regression (G-LR)
Global non-linear	Global decision tree (G-DT)
	Global random forest (G-RF)
Time-sensitive (TS) linear	TS passive-aggressive (TS-PA)
	TS linear support vector (TS-LSV)
	TS logistic regression (TS-LR)
Time-sensitive (TS) non-linear	TS decision tree (TS-DT)
	TS random forest (TS-RF)

The classification rule for aglobal linear learner is:

$$\hat{\mathbf{y}} = \text{sign}(\langle \mathbf{w}_g, \mathbf{x} \rangle), \quad (1)$$

Where $\mathbf{w}_g$is the global weight vector, $\mathbf{x}$ isthe feature vector representation of a tweet, and, $\hat{\mathbf{y}}$ is the prediction. The classification rule for a global non-linear learner is:

$$\hat{\mathbf{y}} = \arg\max_{c \in \{0,1\}} (\varphi_g(c|\mathbf{x})), \quad (2)$$

Where φ_g is a global tree model and $\hat{\mathbf{y}}$ is the predicted class determined as the maximum a posterior (MAP) of the class distribution the $\mathbf{x}$ falls in the leaf.

Time-sensitive (TS) modeling [2] assumes that there are some specific rules within every hour of a day for tweets being retweeted. Each hour in a day corresponds to a local model, therefore, TS modeling consists of one global model (either equation (1) or equation (2)) and 24 local models. The TS linear learner is then:

$$\hat{\mathbf{y}} = \text{sign}\big(\langle \mathbf{w}_g, \mathbf{x} \rangle + \lambda \langle \mathbf{w}_l, \mathbf{x} \rangle\big), \quad (3)$$

Where, $\mathbf{w}_l$ is the local weight vector and λ is the weight that corresponds to the number of tweets that the local model has seen during training, divided by the total number of tweets in the training set. Finally, the TS non-linear learner is defined as:

$$\hat{\mathbf{y}} = \arg\max_{c \in \{0,1\}} (\varphi_g(c|\mathbf{x}) + \lambda\, \varphi_l(c|\mathbf{x})), \quad (4)$$

Where φ_l is a local tree model.

2.2. Learning Methods

Based on previous studies ([3-5], [7, 10]), we propose 5 learning methods: passive-aggresive (PA), linear support vector machine (LSV), logistic regression (LR), decision trees (DT) and random forest-based learning (RF). Combined with the choice for global vs. time-sensitive modeling, this yields a total of 10 approaches; see Table 1.

2.3. Description of Features

Like the learning approaches, the features that we consider are based on a number of previous studies; see Table 2.

Table 2. Features and their origins.

User features	
Number of followers	[15], [3], [4], [6], [7], [10], [12]
Number of friends	[15], [3], [4], [6], [7]
Number of statuses	[15], [3], [7]
Number of favorites	[15], [3]
Number of user listed	[3], [7]
Is a user verified?	[3], [7]
Percentage of replies	[6]
Number of followers / Number of friends	[6]
Average number of tweets per day	[7], [13]
Account age	[15], [7]
Tweet features	
Number of hashtags	[15], [3], [6], [7], [10]
Number of mentions	[15], [3], [6], [7], [10]
Number of URLs	[4], [10]
Length of a tweet	[3], [6], [10]
Novelty score	[3], [3]
Is a tweet a reply?	[3], [7]
Is a tweet a direct message?	[5]
Does a tweet contain a hashtag?	[5]
Does a tweet contain a URL?	[5]
Does a tweet contain '?' or '!'?	[5]

We divide the features into 2 categories as follows.

2.3.1. User Features

A user can possess attributes that make her tweet more likely to be retweeted. We try to capture those attributes from information about the user as follows: how many people follow the user (number of followers), how many people the user follows (number of friends), how many statuses the user has (number of statuses), how many favorite tweets the user has (number of favorites), how many times the user is listed (number of user listed), whether or not the user is verified (is a user verified?). From all tweets authored by a user, we compute the ratio of tweets that have replies to all her tweets (percentage of replies), proportion of number of followers to number of friends (number of followers/ number of friends), how many tweets the user published on average per day (average number of tweets per day), and how old her account age of a user is (in days) when she published the tweet (account age).

2.3.2. Tweet Features

We also extract features about and from tweets themselves. We consider only features that have been shown important in the existing work. The features from tweets are as follows: number of hashtags, mentions, URLs, length of the tweet, the novelty score, whether or not the tweet is a reply, a direct message, whether or not the tweet contains a hashtag, a URL, an exclamation, or question marks. Novelty score is computed as the cosine distance between the TF-IDF vector representations of the tweet and its nearest neighbor tweet published a day before.

3. Experimental Setup

In order to understand what kind of tweets would be retweeted, we created a dataset as follows.We collected lists of Dutch politicians and political journalists from lists curated by *De Issuemakers*, a Dutch communications consultancy company, along with their followers and followees. The total number of politicians and journalists is 304 and total number of followers and followees is around 1.4 million. We collected both tweets and user profiles from September 9 to December 2, 2014 by utilizing the Twitter API. We gathered around 3 million tweets. Our training set comprises tweets from September 9 to December 1 (around 2.7 million tweets) and we use tweets from the last day, December 2nd as our test set (around 0.3 million tweets).

We label a tweet as retweeted or not retweeted by checking whether the tweet has a original status id. If it has one, we acquire the original tweet with the status id and give it label 1. If it has no original status id, we label the tweet 0. We set a threshold value to 2 days to give a tweet a chance to be retweeted. The proportion of retweeted tweets overall is around 33%.

Table 3. The best settings after 5-fold cross-validation

No	Model	Setting
1	G-PA, TS-PA	C= 0.01, loss = squared-hinge
2	G-LSV, TS-LSV	C= 10, dual = false
3	G-LR, TS-LR	C= 10, penalty = l2
4	G-DT, TS-DT	criterion = entropy, splitter = best
5	G-RF, TS-RF	criterion = gini, n_estimators = 30

Table 4. F_1 (%) scores after 5-fold cross-validation on the training set

Model	Average $\pm$ std	Model	Average $\pm$ std
Global models		*TS models*	
G-PA	51.81 ± 0.0229	TS-PA	52.54 ± 0.0160
G-LSV	56.44 ± 0.0010	TS-LSV	56.49 ± 0.0011
G-LR	56.60 ± 0.0011	TS-LR	56.66 ± 0.0011
G-DT	69.18 ± 0.0009	TS-DT	69.75 ± 0.0012
G-RF	74.39 ± 0.0006	TS-RF	75.42 ± 0.0008

Before we run the prediction on the test set, we run 5-fold cross-validation with selected settings [16, 17] for each classifier on our training set. The purpose of cross-validation is to tune all the models and find the best setting from each model. Table 3 describes the best setting resulting from the cross-validation on the training set. The classification performance of our

models after cross-validation is shown in Table 4. We see that the non-linear models (DT, RF) outperform the linear models (PA, LSV, LR), and that time-sensitive modeling only marginally outperforms global modeling on the training set.

For significance testing, we use a one-tailed paired t-test for comparisons between global learners and between time-sensitive learners; significant differences are marked using $^{\uparrow}$ for significant differences at $\alpha = 0.01$.We use McNemar's test to measure the significance differences in prediction accuracy of the global learners and the TS learners.

4. Results

4.1. Prediction Accuracy

We run all models from Table 1. Table 5 shows the F_1 score of the models with user features only, tweet features only, and with both of them on the test set. We see that the global and time-sensitive random forest model (G-RF) and (TS-RF) achieve the highest performance. We also see that user features outperform tweet features and that their union outperforms both.

4.2. Feature Selection

In order to understand the overall contribution of each individual feature for prediction accuracy, we utilize recursive feature elimination (RFE) and compute gini importance on all features in the global linear and non-linear models. RFE starts by training models with all features. The feature whose absolute weight is then found smallest is pruned from the set; RFE continues like this recursively until there is only one feature left. This last feature is the first-ranked feature in Table 6. Measuring the importance of a feature in decision tree or random forest equals computing the decrease of impurity of the nodes over all trees in the forest [16]. The lower decreasing is higher the importance of the feature in the decision tree or random forest.

Table 5. Comparison of F_1 (%) scores from different classifiers and feature sets on the test set. In the rightmost column, statistically significant differences with the previous row (in the same part of the table) are marked with $^{\uparrow}$

Model	User features	Tweet features	Both
Global models			
G-PA	50.81	30.26	51.58
G-LSV	51.48	18.90	$54.65^{\uparrow}$
G-LR	55.18	23.01	$58.55^{\uparrow}$
G-DT	68.01	46.39	$67.98^{\uparrow}$
G-RF	69.63	38.78	$\mathbf{73.66}^{\uparrow}$
TS models			
TS-PA	51.15	30.31	51.84
TS-LSV	51.52	18.98	$54.66^{\uparrow}$
TS-LR	55.18	23.05	$58.59^{\uparrow}$
TS-DT	68.08	46.39	$67.97^{\uparrow}$
TS-RF	71.06	46.32	$\mathbf{74.47}^{\uparrow}$

Table 6. The top-5-feature rankings generated from RFE for global passive-aggressive (G-PA), global linear support vector (G-LSV), and global logistic regression (G-LR)

Rank	G-PA	G-LSV	G-LR
1	Number of user listed	Number of user listed	Number of user listed
2	Is a user verified?	#followers	#followers
3	#followers	Average #tweets/day	Average #tweets/ day
4	Average #tweets / day	#followers / #friends	#followers / #friends
5	#followers / #friends	Length of a tweet	Is a user verified?

#tweets = number of tweets, #followers = number of followers, #friends = number of friends

Table 6 shows that the number of times a user is listed, the number of followers, and the average number of tweets published per day are the features that contribute most to the prediction accuracy. Moreover, the important features of our global random forest model in Table 7 are similar to the ones identified by RFE for the linear learners.

Table 7. The top-5-feature importance rankings for global decision tree classifier (G-DT) and global random forest classification (G-RF) based on gini importance

Rank	G-DT	G-RF
1	Number of user listed	Number of user listed
2	Average tweets per day	Number of followers
3	Novelty	Average tweets per day
4	Average tweets per day	Number of followers / Number of friends
5	Account age	Novelty

Interestingly, both graphs in Figure 1 show that there is more than 10% increase of F_1 score (blue line) when we add the feature, number of tweets published. We find that G-LSV and G-LR with selecting 4 best features plus ”number of tweets published” can achieve performance comparable with the performance of both models utilizing all the features (green dashed line in Figure 1).

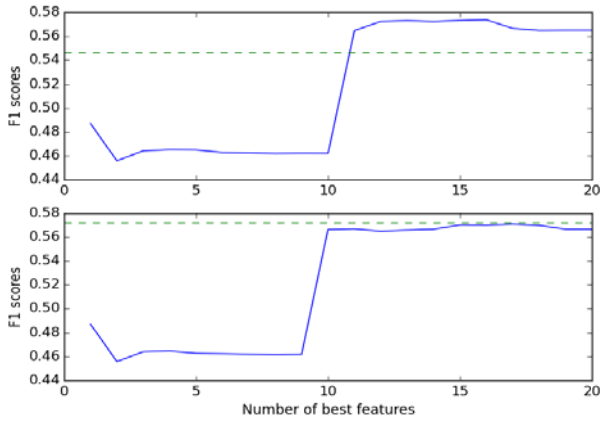

Figure 1. (Top): F_1 scores of G-LSV trained incrementally with more features (x-axis). (Bottom): F_1 scores of G-LR. F_1 scores are computed from 5-fold cross-validation, the order of the features is determined by RFE computed from the corresponding model. The dashed green line depicts the F_1 score of the corresponding model trained on the best 4 features + the number of tweets published (the feature causing the boost in F_1 score for both models)

4.3. Time-sensitive Modeling

Time-sensitive modeling generally yields better performance than the global models; the exception is G-DT, which outperforms TS-DT by 0.01%. However, the McNemar significance test does not indicate that any of the differences are significant.

We also carried out experiments to find the optimal global λ in equation (3) and (4) as trying individual λ for each local model is intractable. However, the local model inside time-sensitive modeling is still unable to contribute to help improve the predictions.

We conclude that, on our dataset and unlike the findings by Petrovic, et al., [3], time-sensitive modeling has a very limited contribution to the overall performance the use of a strong learning model is far more important.

5. Conclusion

We provide a study and comparison of retweet prediction approaches. To the best of our knowledge, this is the first attempt to collect best performing features and contrast linear and non-linear learning methods. Specifically, we answer three research questions. We demonstrate that, on our dataset, a random forest-based learning method, which has not been employed in previous studies, outperforms all other learning methods that we consider. We find that user features are more important than tweet features in making correct predictions and that the best three features are: number of times a user is listed, number of followers, and average number of tweets published per day. Using feature selection, we find that the number of tweets

published combined with the four best features leads to performance levels of G-LSV and G-LR models that are comparable using all features. Lastly, time-sensitive modeling has limited benefits on our dataset.

As to future work, we plan to study the potential of signals inferred from external sources (such as news or Wikipedia) for retweet prediction.

References

[1] Goyal S. Facebook, Twitter, Google+: Social Networking. *International Journal of Social Networking and Virtual Communities*. 2012; 1(1).
[2] Kwak H, Lee C, Park H, Moon S. *What is Twitter, a Social Network or a News Media?* Proceedings of the 19th International Conference on World Wide Web. Raleigh, North Carolina, USA. 2010: 591-600.
[3] Petrovic S, Osborne M, Lavrenko V. *RT to Win! Predicting Message Propagation in Twitter*. Proceedings of the Fifth International Conference on Weblogs and Social Media. Barcelona, Spain. 2011.
[4] Hong L, Dan O, Davison BD. *Predicting Popular Messages in Twitter*. International World Wide Web Conferences. Hyderabad, India. 2011: 57-58.
[5] Naveed N, Gottron T, Kunegis J, Alhadi AC. *Bad News Travel Fast: A Content-based Analysis of Interestingness on Twitter*. Web Science Conference. Koblenz, Germany. 2011: 8.
[6] Artzi Y, Pantel P, Gamon M. *Predicting Responses to Microblog Posts*. Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics Proceedings. Montreal, Canada. 2012: 602-606.
[7] Xu Z, Yang Q. *Analyzing User Retweet Behavior on Twitter*. The International Conference on Advances in Social Network Analysis and Mining. Calgary, Canada. 2012: 46-50.
[8] Caruana R, Niculescu-Mizil A. *An Empirical Comparison of Supervised Learning Algorithms*. Proceedings of the 23rd International Conference on Machine Learning. Pittsburgh, USA. 2006: 161-168.
[9] Fernandez-Delgado M, Cernadas E, Barro S, Amorim D. Do we need hundreds of classifiers to solve real world classification problems?*The Journal of Machine Learning Research*. 2014; 15(1): 3133-3181.
[10] Jenders M, Kasneci G, Naumann F. *Analyzing and Predicting Viral Tweets*. International World Wide Web Conferences. Rio de Janeiro, Brazil. 2013: 657-664.
[11] Gao S, Ma J, Chen Z. *Modeling and Predicting Retweeting Dynamics on Microblogging Platforms*. Proceedings of the Eighth ACM International Conference on Web Search and Data Mining. New York City, USA. 2015:107-116.
[12] Zaman T, Fox EB, Bradlow ET. A Bayesian Approach for Predicting the Popularity of Tweets. *The Annals of Applied Statistics*. 2014; 8(3): 1583-1611.
[13] Macskassy SA, Michelson M. *Why do People Retweet? Anti-homophily Wins the Day!* Proceedings of the Fifth International Conference on Weblogs and Social Media. Barcelona, Spain. 2011.
[14] Morchid M, Dufour R, Bousquet P-M, Linares G, Torres-Moreno J-M. Feature Selection using Principal Component Analysis for massive retweet detection. *Pattern Recognition Letters*. 2014; 49: 33-39.
[15] Suh B, Hong L, Pirolli P, Chi EH. *Want to be Retweeted? Large Scale Analytics on Factors Impacting Retweet in Twitter Network*. 2010 IEEE Second International Conference on Social Computing (SocialCom). Minneapolis, USA. 2010: 177-184.
[16] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*. 2011; 12: 2825-2830.
[17] Wang L. Machine Learning in Big Data. *International Journal of Advances in Applied Science*. 2016: 4(4).
[18] Breiman L. Random Forests. *Machine Learning*. 2001; 45(1): 5-32.