# Table of Contents

## Regular Papers

**Data Selection and Fuzzy-Rules Generation for Short-Term Load Forecasting Using ANFIS**  
Mamunu Mustapha, Mohd Wazir Mustafa, Saiful Nizam Abd. Khalid  
791

**Temperature Control of the 25kW Parabolic Dish-Stirling Engine System**  
Liaw Geok Pheng, Mohd Ruddin Ab Ghani, Chin Kim Gan, Tole Sutikno  
800

**Sensitivity Analysis and Comparison between 25 kW Parabolic Dish System**  
Mohd Ruddin Ab Ghani, Liaw Geok Pheng, Chin Kim Gan, Tole Sutikno  
807

**Transformer Fault Diagnosis Method Based on Dynamic Weighted Combination Model**  
Hongli Yun, Run Liu, Linjian Shangguan  
815

**Research on Optimization Strategy to Data Clustered Storage of Consistent Hashing Algorithm**  
Ningia Qiu, Xiaojuan Hu, Peng Wang, Huamin Yang  
824

**A Comprehensive Test Approach on High-Power Low-Noise Intermodulation Distortion**  
Lei Wang, Jingyi Zhang  
831

**Arduino Based Paperless Queue Management System**  
Aiman Zakwan Jidin, Norfazliza Mohd Yusof, Tole Sutikno  
839

**Internet Protocol Based Satellite On-Board System**  
Emir Husni, Nazmi Febrian, Angga Putra  
846

**Design and Fabrication of Compact MEMS Electromagnetic Micro-Actuator with Planar Micro-Coil Based on PCB**  
Roer Eka Pawinanto, Jumril Yunas, Burhanuddin Majlis, Azrul Hamzah  
856

**Critical Condition in CuInAlSe2 Growth of Solar Cell Absorber**  
867

**The Design and Performance Analysis of DTMB System**  
Emy Haryati, Tubagus Maulana Kusuma, Busono Soerowidjo, Purnawarman Musa  
873

**Modeling and Simulation of Erbium doped Photonic Crystal Fiber**  
Farah Diana Mahad, Abu Sahmah M.Supa‘at, David Ian Forsyth, Tong Sun, Asrul Izam Azmi  
880

**Adaptive Resource Allocation Algorithm in Wireless Access Network**  
Zhanjun Liu, Yue Shen, Zhonghua Yu, Fengxie Qin, Qianbin Chen  
887

**Error Resilient Multipath Video Delivery on Wireless Overlay Networks**  
Uma Maheswari, Sudarshan TSB  
894

**Towards Smooth and High-Quality Bitrate Adaptation for HTTP Adaptive Streaming**  
Lihong Geng, Liang Pan, Yiqiang Sheng, Zhichuan Guo  
904
Distributed Clustering Based on Node Density and Distance in Wireless Sensor Networks
Sasikumar P, Shankar T, Sibaram Khara

Optimization of Hydrogen-fueled Engine Ignition Timing Based on L-M Neural Network Algorithm
Lijun Wang, Yuan Liu, Yahui Liu, Wei Wang, Yanan Zhao, Zhenzhong Yang

Nurses Scheduling by Considering the Qualification using Integer Linear Programming
Maya Widyastiti, Amril Aman, Toni Bakhtiar

Prediction Model of Smelting Endpoint of Fuming Furnace Based on Grey Neural Network
Song Qiang, WU Yaochun

Application of Nonlinear Dynamical Methods for Arc Welding Quality Monitoring
Shuguang Wu, Yiqing Zhou

Recognition of Odor Characteristics Based on BP Neural Network
Wu Lei, Fang Jiandong, Zhao Yudong

H-Infinity Control for Pitch-Roll AR.Drone
Agung Prayitno, Veronica Indrawati, Clark Arron

Compressive Sensing Algorithm for Data Compression on Weather Monitoring System
Rika Sustika, Bambang Sugiarto

Fuzzy C-Means Clustering Based on Improved Marked Watershed Transformation
Cuijie Zhao, Hongdong Zhao, Wei Yao

A Technique to Improve Ridge Flows of Fingerprint Orientation Fields Estimation
Saparudin, Ghazali Sulon

Ventricular Tachyarrhythmia Prediction based on Heart Rate Variability and Genetic Algorithm
Khang Hua Boon, Malarvili Bala Krishnan, Mohamed Khalil-Hani

GPU CUDA accelerated Image Inpainting using Fourth Order PDE equation
Edwin Prananta, Pranowo, Djoko Budianto

Recognition of Fission Signals Based on Wavelet Analysis and Neural Network
Li Li, Liu Keqi, Hu Gen

Musical Genre Classification Using SVM and Audio Features
Achmad Benny Mutiara, Rina Refianti, Nadia R.A. Mukarromah

An Improved Adaptive Niche Differential Evolution Algorithm
Hui Wang, Changtong Song

Particle Swarm Optimization Performance: Comparison of Dynamic Economic Dispatch with Dantzig-Wolfe Decomposition
Mohd Ruddin Ab Ghani, Saif Tahseen Hussein, Zanariah Jano, Tole Sutikno

A Comparison of Retweet Prediction Approaches: The Superiority of Random Forest Learning Method
Hendra Bunyamin, Tomas Tunys
Hadoop Performance Analysis on Raspberry Pi for DNA Sequence Alignment
   Jaya Sena Turana, Heru Sukoco, Wisnu Ananta Kusuma 1059

A Sentiment Knowledge Discovery Model in Twitter’s TV Content Using Stochastic Gradient Descent Algorithm
   Lira Ruhwinaningsih, Taufik Djaatna 1067

Hybrid Hierarchical Collision Detection Based on Data Reuse
   Jiancai Hu, Kejing He, Xiaobin Lin, Funan Lin 1077

Big Data Analysis with MongoDB for Decision Support System
   Sulistyo Heripracoyo, Roni Kurniawan 1083

Brightness and Contrast Modification in Ultrasonography Images Using Edge Detection Results
   Retno Supriyanti, Suwitno, Yogi Ramadhan, Haris Budi Widodo,
   Tutik Ida Rosanti 1090

An Improved Artificial Bee Colony Algorithm for Staged Search
   Shoulin Yin, Jie Liu, Lin Teng 1099

MRI Sagittal Image Segmentation from Patients with Abdominal Aortic Aneurysms
   Desti Riminarsih, Cut Maisyarah Karyati, Achmad Benny Mutiara,
   Bambang Wahyudi, E. Ernastuti 1105

Multi-Criteria in Discriminant Analysis to Find the Dominant Features
   Arif Muntasir, Indah Agustien Siradjuddin, Rima Tri Wahyuningrum 1113

MapReduce Integrated Multi-algorithm for HPC Running State Analysis
   ShuRen Liu, ChaoMin Feng, HongWu Luo, Ling Wen 1123

Classification of Motorcyclists not Wear Helmet on Digital Image with Backpropagation Neural Network
   Sutikno, Indra Waspada, Nurdin Bahtiar, Priyo Sidik Sasonko 1128

Quasi-Newton Method for Absolute Value Equation Based on Upper Uniform Smoothing Approximation Function
   Longquan Yong, Shouheng Tuo 1134

Flow Fair Sampling Based on Multistage Bloom Filters
   Liu Yuanzhen, Huang Shurong, Liu Jianzhao 1142

A New Semi-supervised Clustering Algorithm Based on Variational Bayesian and Its Application
   Shoulin Yin, Jie Liu, Lin Teng 1150

Metamorphic Malware Detection Based on Support Vector Machine Classification of Malware Sub-Signatures
   Ban Mohammed Khammas, Alireza Monemi, Ismahanani Ismail,
   Sulaiman Mohd Nor, M.N. Marsono 1157

Wireless Sensor Network Design based on Hybrid Tree-Like Mesh Topology as a New Platform for Air Pollution Monitoring System
   Muhammad Iqbal, Muhammad Fuad, Heru Sukoco, Husin Alatas 1166

Image Retrieval Based on Multi Structure Co-occurrence Descriptor
   Agus Eko Minarno, Arrie Kurniawardhani, Filtri Bimantoro 1175
A Comparison of Retweet Prediction Approaches: The Superiority of Random Forest Learning Method

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

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 have 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 mostly 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

Copyright © 2016 Universitas Ahmad Dahlan. All rights reserved.

1. Introduction

Social media like Twitter has provided a platform for spreading information among users \cite{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., \cite{3-7}. In addition, different modeling setups have been proposed; e.g., Petrovic, et al., \cite{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., \cite{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.
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 our 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 followers, 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>
<thead>
<tr>
<th>Group of learners</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global linear</td>
<td>Global passive-aggressive (G-PA)</td>
</tr>
<tr>
<td></td>
<td>Global linear support vector (G-LSV)</td>
</tr>
<tr>
<td></td>
<td>Global logistic regression (G-LR)</td>
</tr>
<tr>
<td>Global non-linear</td>
<td>Global decision tree (G-DT)</td>
</tr>
<tr>
<td></td>
<td>Global random forest (G-RF)</td>
</tr>
<tr>
<td>Time-sensitive (TS) linear</td>
<td>TS passive-aggressive (TS-PA)</td>
</tr>
<tr>
<td></td>
<td>TS linear support vector (TS-LSV)</td>
</tr>
<tr>
<td></td>
<td>TS logistic regression (TS-LR)</td>
</tr>
<tr>
<td>Time-sensitive (TS) non-linear</td>
<td>TS decision tree (TS-DT)</td>
</tr>
<tr>
<td></td>
<td>TS random forest (TS-RF)</td>
</tr>
</tbody>
</table>

A Comparison of Retweet Prediction Approaches: The Superiority... (Hendra Bunyamin)
The classification rule for a global linear learner is:

$$\hat{y} = \text{sign}(\langle w_g, x \rangle),$$  \hspace{1cm} (1)$$

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

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

Where $\varphi_g$ is a global tree model and $\hat{y}$ is the predicted class determined as the maximum a posteriori (MAP) of the class distribution the $x$ 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:

$$\hat{y} = \text{sign}(\langle w_g, x \rangle + \lambda \langle w_i, x \rangle),$$  \hspace{1cm} (3)$$

Where, $w_i$ is the local weight vector and $\lambda$ 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:

$$\hat{y} = \arg\max_{c \in \{0,1\}} (\varphi_g(c|x) + \lambda \varphi_i(c|x)),$$  \hspace{1cm} (4)$$

Where $\varphi_i$ is a local tree model.

2.2. Learning Methods

Based on previous studies ([3-5], [7, 10]), we propose 5 learning methods: passive-aggressive (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>
<thead>
<tr>
<th>Table 2. Features and their origins.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>User features</strong></td>
</tr>
<tr>
<td>Number of followers</td>
</tr>
<tr>
<td>Number of friends</td>
</tr>
<tr>
<td>Number of statuses</td>
</tr>
<tr>
<td>Number of favorites</td>
</tr>
<tr>
<td>Number of user listed</td>
</tr>
<tr>
<td>Is a user verified?</td>
</tr>
<tr>
<td>Percentage of replies</td>
</tr>
<tr>
<td>Number of followers / Number of friends</td>
</tr>
<tr>
<td>Average number of tweets per day</td>
</tr>
<tr>
<td>Account age</td>
</tr>
<tr>
<td><strong>Tweet features</strong></td>
</tr>
<tr>
<td>Number of hashtags</td>
</tr>
<tr>
<td>Number of mentions</td>
</tr>
<tr>
<td>Number of URLs</td>
</tr>
<tr>
<td>Length of a tweet</td>
</tr>
<tr>
<td>Novelty score</td>
</tr>
<tr>
<td>Is a tweet a reply?</td>
</tr>
<tr>
<td>Is a tweet a direct message?</td>
</tr>
<tr>
<td>Does a tweet contain a hashtag?</td>
</tr>
<tr>
<td>Does a tweet contain a URL?</td>
</tr>
<tr>
<td>Does a tweet contain '?' or '?'</td>
</tr>
</tbody>
</table>

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 an 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>
<thead>
<tr>
<th>Table 3. The best settings after 5-fold cross-validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>No</td>
</tr>
<tr>
<td>-----</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 4. F1(%) scores after 5-fold cross-validation on the training set</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
</tr>
<tr>
<td>-------</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>G-PA</td>
</tr>
<tr>
<td>G-LSV</td>
</tr>
<tr>
<td>G-LR</td>
</tr>
<tr>
<td>G-DT</td>
</tr>
<tr>
<td>G-RF</td>
</tr>
</tbody>
</table>

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 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 F1 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 F1 (%) 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

<table>
<thead>
<tr>
<th>Model</th>
<th>User features</th>
<th>Tweet features</th>
<th>Both</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global models</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G-PA</td>
<td>50.81</td>
<td>30.25</td>
<td>51.58</td>
</tr>
<tr>
<td>G-LSV</td>
<td>51.40</td>
<td>18.90</td>
<td>54.65*</td>
</tr>
<tr>
<td>G-LR</td>
<td>55.18</td>
<td>23.01</td>
<td>58.55*</td>
</tr>
<tr>
<td>G-DT</td>
<td>68.01</td>
<td>46.39</td>
<td>70.01*</td>
</tr>
<tr>
<td>G-RF</td>
<td>69.63</td>
<td>38.78</td>
<td>73.66*</td>
</tr>
<tr>
<td>TS models</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TS-PA</td>
<td>51.15</td>
<td>30.31</td>
<td>51.84</td>
</tr>
<tr>
<td>TS-LSV</td>
<td>51.52</td>
<td>18.98</td>
<td>54.66*</td>
</tr>
<tr>
<td>TS-LR</td>
<td>55.18</td>
<td>23.05</td>
<td>58.59*</td>
</tr>
<tr>
<td>TS-DT</td>
<td>68.08</td>
<td>46.39</td>
<td>70.97*</td>
</tr>
<tr>
<td>TS-RF</td>
<td>71.06</td>
<td>46.32</td>
<td>74.47*</td>
</tr>
</tbody>
</table>

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).

<table>
<thead>
<tr>
<th>Rank</th>
<th>G-PA</th>
<th>G-LSV</th>
<th>G-LR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Number of user listed</td>
<td>Number of user listed</td>
<td>Number of user listed</td>
</tr>
<tr>
<td>2</td>
<td>Is a user verified?</td>
<td>#followers</td>
<td>#followers</td>
</tr>
<tr>
<td>3</td>
<td>#followers</td>
<td>Average #tweets/day</td>
<td>Average #tweets/day</td>
</tr>
<tr>
<td>4</td>
<td>Average #tweets/day</td>
<td>#followers / #friends</td>
<td>#followers / #friends</td>
</tr>
<tr>
<td>5</td>
<td>#followers / #friends</td>
<td>Length of a tweet</td>
<td>Is a user verified?</td>
</tr>
</tbody>
</table>

#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

<table>
<thead>
<tr>
<th>Rank</th>
<th>G-DT</th>
<th>G-RF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Number of user listed</td>
<td>Number of user listed</td>
</tr>
<tr>
<td>2</td>
<td>Average tweets per day</td>
<td>Number of followers</td>
</tr>
<tr>
<td>3</td>
<td>Novelty</td>
<td>Average tweets per day</td>
</tr>
<tr>
<td>4</td>
<td>Average tweets per day</td>
<td>Number of followers / Number of friends</td>
</tr>
<tr>
<td>5</td>
<td>Account age</td>
<td>Novelty</td>
</tr>
</tbody>
</table>

Interestingly, both graphs in Figure 1 show that there is more than 10% increase of F₁ 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).

![Graph showing F₁ score incrementality with more features](image)

Figure 1. (Top): F₁ scores of G-LSV trained incrementally with more features (x-axis). (Bottom): F₁ scores of G-LR. F₁ 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 F₁ score of the corresponding model trained on the best 4 features + the number of tweets published (the feature causing the boost in F₁ 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

---

**A Comparison of Retweet Prediction Approaches: The Superiority… (Hendra Bunyamin)**
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


