TAF Journal of Advances in Technology and Engineering Research JATER
2019, **5**(5): 219-226

PRIMARY RESEARCH

Personalized spoiler detection in tweets by using support vector machine

Isao Sasano [1*], Kei Morisawa [2], Yutaka Hirakawa [3]
[1, 2, 3] Department of Computer Science and Engineering, Shibaura Institute of Technology, Tokyo, Japan

Keywords
Spoiler detection
TV program tweet
SVM

Received: 9 May 2019
Accepted: 11 July 2019
Published: 31 October 2019

Abstract
At present, numerous people watch prerecorded TV programs as daily leisure. Concerning soap operas or sports, the viewers may not want to be informed about the results before watching the programs; however, they may check tweets on devices, such as smartphones, which can accidentally include contents referring to spoilers. To avoid reading such content, several approaches were proposed to detect spoilers in texts (both long and short ones), including tweets. In the study by Jeon et al. focused on detecting spoilers in tweets, only one person attached labels to tweets, and the labeled tweets were used to train detectors. The trained detector was tuned for one person and, therefore, could be unsuitable for others. A tweet published in the middle of a baseball game can be considered a spoiler by some people and not by others; therefore, a personalized detection method is preferred. However, to the best of our knowledge, none of the related studies has considered such a personalized approach. To address this problem, we propose a semi-supervised approach to detect spoilers in tweets using a support vector machine (SVM) in which each user attaches labels to tweets. After that, SVM executes the same procedure for other unlabeled tweets through bootstrapping. To verify the suitability of the proposed approach to personalize detectors, we conducted an experiment in which two participants were asked to attach labels to tweets. The experimental results indicate that this approach is efficient for personalized detection based on the Mann-Whitney U test.

I. INTRODUCTION

At present, many people prefer watching prerecorded TV programs as their daily leisure. Concerning soap operas or sports, the viewers may not want to be informed about the results prior to watching the programs; however, they may check comments or tweets on devices, such as smartphones, to communicate with their acquaintances. Recently, smartphones have been widely used all over the world. Along with smartphones, Social Networking Services (SNS), such as LINE and Twitter, have become popular. Using SNS, people can describe their recent activities, share their feelings on their daily lives, and so on. Comments on SNS frequently refer to feelings about TV programs, some of which may be related to the contents of soap operas, results of sports, and other similar posts. Although it is entertaining for people to exchange comments for sharing for sharing their feelings on TV programs, particular comments may be considered as spoilers by people who do not watch TV programs in real-time. Hereinafter, we denote such comments as spoilers. Many people consider spoilers as frustrating, and in some cases, this may even cause quarrels between users. However, deciding on what kinds of comments correspond to spoilers mainly depends on personal opinions.
To avoid reading such contents, several approaches were proposed to detect spoilers in texts (both long and short ones), including tweets. In the study by Jeon et al. [1] focused on detecting spoilers in tweets, only one person attached labels to tweets, and the labeled tweets were used to train detectors. Although the trained detector could be specialized about a particular person, the authors did not consider whether the same trained detector was suitable for other persons. In general, a tweet published in the middle of a baseball game can be considered as a spoiler by some people and as acceptable content for others; therefore, a

*Corresponding author: Isao Sasano
†email: sasano@sic.shibaura-it.ac.jp

personalized detection method is required. However, to the best of our knowledge, none of the current research work has presented such a personalized approach.

In the present study, we focus on the tweets published on Twitter that refer to the TV programs dedicated to professional baseball games in Japan. We propose a personalized approach to identify whether or not a tweet in question can be considered as a spoiler. Herein, similarly to the study by Jeon et al. [1], we employ a semi-supervised approach [2] to detect spoilers in tweets using a SVM [3, 4], implying that each user attaches labels to several tweets which are then inputted into the machine as the training data. Thereafter, the SVM attaches the labels to the unlabeled tweets. Labeling is performed iteratively, with several unlabeled tweets being labeled at once. To verify that the proposed approach is suitable to personalize detectors, we conduct an experiment in which two experiment participants are asked to attach labels to tweets. The experimental results confirm that the proposed approach can be efficiently applied to personalized detection according to the Mann-Whitney U-test [5, 6].

The rest of the paper is organized as follows. Section II describes the proposed approach. Section III provides the details on the implementation and preprocessing of the dataset. Section IV describes the experimental setup and results. Section V represents the testing of the obtained experimental results. Section VI discusses the related research works. Section VII concludes the paper and describes future work.

II. THE PROPOSED APPROACH

To identify whether or not a tweet can be considered as a spoiler, we construct an approach based on SVM that can be used to classify tweets into two classes: spoilers and non-spoilers. According to the study [1], we apply a semisupervised approach to detect spoilers in tweets using SVM, implying that each user attaches labels to several tweets. If we used supervised learning, each user had to attach labels to a large number of tweets, such as 1,000, which would be burdensome to users and, therefore, difficult to use.

III. IMPLEMENTATION AND PREPROCESSING OF THE DATASET

We have implemented the proposed system based on the approach presented in Section II by using the Python language and utilizing the Python library sci-kit-learn. In this section, we describe the dataset (the set of tweets) we use, preprocessing of the dataset, vectorizing the preprocessed tweets, and the construction of SVM.

A. Dataset

As a training and testing dataset, we employed the collection of tweets regarded to professional baseball games of Yokohama DeNA Baystars in Japan published between September 11, and October 10 in 2018. Overall, we collected 10,000 tweets for 20 games, namely, 500 tweets per game. While collecting tweets dedicated to a game, we divided the games into five periods and collected 100 tweets for each period.

About concerning the tweets concerning Yokohama DeNA Baystars, a hashtag baystars could be attached, and some other hashtags could also be attached. However, it depended on users whether or not to attach this hashtag. Users may not attach any hashtag. To detect spoilers in the tweets without hashtags, it was necessary to analyze the contents of tweets. In the present study, we assumed that the difference between the tweets with hashtags and those without them was insignificant. Therefore, we collected only the tweets with the hashtag baystars. Although this assumption may not hold, we considered that the experiments should be conducted with an appropriate dataset in the first place and considered the analysis of the contents of tweets as a separate issue.

We used the Python library Tweepy [7] to collect the tweets in question. We saved the tweets collected per game in a separate Excel file. Each Excel file contained the contents of the tweets in column A and the dates of publishing the tweets in column B; the hashtags were deleted from ree tweet's content.

We provided the Excel files to the two experiment participants and asked them to indicate 1 or 2 in column C, where one corresponded to a spoiler and 2 to a non-spoiler. The experiment participants were university students aged between 20 and 22. In Table 1, we represent a part extracted from the Excel files, in which column A contains the contents of tweets written in Japanese. It should be noted that the target audience does not have to understand the meaning of the contents of tweets in column A since the classification in column C depends on users.

B. Vectorizing Tweets

To construct an SVM, it was necessary to vectorize each tweet. To do that, we firstly performed morphological analysis on the Japanese language by using the Python library Janome [8] and then preprocessed the sequence of words as follows:

- Deleting URLs and replies
- Deleting numerals and symbols
- Transforming each word into the one in the original form

ISSN: 2414-4592
DOI: 10.20474/jater-5.5.4

• Transforming English alphabet symbols into the lowercase and single-byte characters
• Transforming Japanese characters including Katakana into the double-byte characters
• Replacing sequences consisting of the same symbol with the single symbol: for example, replacing !!!! with !.

Besides the preprocessing steps mentioned above, we delete the words of high and low frequencies. After preprocessing, we vectorize each tweet as a Bag of Words (BoW) [9] that is defined as a vector containing the frequency of each word and then, simplify each obtained vector into a vector in 64 dimensions by applying the Latent Semantic Indexing (LSI) [10].

With regard to deleting numerals, we note that as numerals may represent the score of games, it is deemed natural to leave numerals. However, as a result of the conducted experiments, we observe that deleting numerals allows increasing the accuracy of the classification. With regard to deleting symbols, they were deleted as we consider that they do not represent any meaning by themselves.

The aforementioned preprocessing steps are described in detail below:

• Deleting URLs and replies. We delete URLs and replies through regular expression matching. URLs are defined as the strings starting with http:// or https://. Replies are messages written as comments to the messages published by other users. Replies start with @ that is followed by alpha-numeric characters.
• Transforming each word into the one in the original form. We transform words into the ones in the original forms by using the Python library Janome [8].
• Replacing the sequences consisting of the same symbol with the single symbol.

Users of Twitters in Japan typically repeat w or repeat a vowel letter that is the last character of a word to indicate.

TABLE 1
EXTRACTED PART OF THE EXCEL FILE COMPOSED OF TWEETS

Tweet	Date	Classification
つっつは守備で無理しないで^^;	18/09/19	2
東4イニングパーフェクト	18/09/19	1
クワが取るかと思ったw 東先発で張り切ってる筒香お兄さんなのかなww	18/09/19	2
東克樹が今日も躍動している	18/09/19	2

TABLE 2
RELATION BETWEEN THE DIMENSION AND THE ACCURACY OF CLASSIFICATION

Dimeniosn	F-Measure
2	0.571
4	0.615
8	0.631
16	0.659
32	0.692
64	0.702
128	0.691

The excitation or the disappointment. We normalize such sequences by transforming them into a single character to process them accordingly.
• Deleting words of high and low frequencies.

To do this, we use the Python library gensim [11]. The words of high and low frequencies are not considered to contribute to classifying the tweets. We delete the words that appear in more than 80% of the considered collection of tweets and those that appear in tweets less than five times.

After preprocessing, we denote the remaining words as the characteristic ones. In the case when no words remain in a tweet, we do not consider this tweet for learning. By vectorizing a tweet by BoW, we describe the resulting words as

ISSN: 2414-4592
DOI: 10.20474/jater-5.5.4

vectors in the dimension of the number of the characteristic words in the tweet. In typical cases, the dimension of a vector obtained by BoW becomes large, so that if we used the vector without any change, constructing SVM might require high performance computers. Therefore, we decrease the dimension by applying LSI. In the experiment, we considered dimensions between 2 and 128 and finally adopted 64, which allowed achieving the best accuracy, as shown in Table 2.

C. Construction of SVM

While constructing SVM, we employed the semi-supervised approach (also called bootstrapping), implying that the machine attached labels to unlabeled tweets after training, during which labeled and unlabeled tweets were inputted into the SVM. In the bootstrapping, we consider the following iterations:
(1) Learning by using the labeled tweets only
(2) Attaching labels to several unlabeled tweets
(3) Going back to (1).
When all tweets are labeled or when the number of iterations reaches a fixed amount of times, we stop executing the algorithm. We employ semisup-learn [12], which is a semisupervised learning framework for Python using scikit-learn classifiers for the partially labeled input data. We set the threshold as 80%, and the maximum repeating time as 800.

IV. EXPERIMENTS

To evaluate the proposed approach, we conducted the experiments to analyze two points: whether the user burden related to labeling tweets was decreased and whether classification of tweets was personalized.

A. Decreasing the user Burden

In the experiment, we changed the ratio of the labeled tweets between 10% and 100%, in 10% increments, considering all tweets, including the unlabeled ones. In the case of the ratio equal to 100%, we performed the usual supervised learning.
Following the study by Jeon et al. [1], we evaluated the detector in terms of the F-measure value (also called F-score), which is the harmonic mean of the precision and recall. We represented the precision, recall, and F-measures in Table 3. For each ratio, we applied SVM 10 times, computed the above three values, and estimated their average.
The outcomes presented in Table 3 indicated that the case when 50% of the tweets were labeled achieved the best result. The case of 100% achieved the result worse than those corresponding to the case of 50%. It took approximately 15 hours for each experiment participant to attach labels to 10,000 tweets. As a result, in the long-time experiment, the experiment participants might not recall which labels rigidly they had attached to similar tweets that were borderline between spoilers and non-spoilers.

TABLE 3
PRECISION, RECALL, AND F-MEASURE OF THE CLASSIFICATION

The Ratio of Labeled Data	Percision	Recall	F-Measure
10%	0.654	0.656	0.650
20%	0.717	0.718	0.716
30%	0.717	0.717	0.717
40%	0.718	0.718	0.717
50%	0.740	0.741	0.739
60%	0.733	0.733	0.733
70%	0.735	0.735	0.734
80%	0.730	0.731	0.730
90%	0.727	0.728	0.727
100%	0.727	0.728	0.727

TABLE 4
EXTRACTED PART OF THE TEST DATA IN EXCEL, WHERE THE FIRST COLUMN CONTAINS THE TWEETS IN JAPANESE

Tweet	Date	Classification by A	Classification by B
荒波の登場曲に合わせてタオル回すの楽しかったなぁ...荒波翔	18/10/01	2	2
調子悪くても三振はとれるのよね	18/10/01	2	1
@YB NIKKO二回無失点	18/10/01	1	1
高山久しぶりやな	18/10/01	2	2

ISSN: 2414-4592
DOI: 10.20474/jater-5.5.4

B. Personalization

We consider that SVM A is the one that was constructed using the tweets labeled by the experiment participant A, and SVM B is the one labeled by the experiment participants B. We also prepared the two sets of the test data, which were the sets of tweets labeled by the experiment participants A and B. Here, we employed the same set of (unlabeled) tweets for the experiment participants A and B. A part of the test data is provided in Table 4.

After the learning, we checked whether SVM A was specialized for the experiment participant A by considering the two sets of the test data. Labels were attached by the experiment participants A and B, respectively. We checked SVM B similarly.

We provide the obtained results in Table 5, which shows the averages of the values of F-measure in Table 6 and Table 7 that appears in Section V. They indicated that the classification results corresponding to the test data by SVM A were closer to those of the classification done by A and that the classification results corresponding to the test data by SVM B were closer to those obtained by B. In the next section, we describe testing on whether these results have a statistically significant difference.

V. TESTING

In this section, we describe the testing on whether or not the results presented in Section IV have statistically significant difference. As the data obtained in the experiments in Section IV did not follow the normal distribution, we used the Mann- Whitney U test [5].

We let the group A be the F-measure values obtained by using the test data (tweets) labeled by the experiment participant A for ten-time experiments, where the ratio of the labeled data is 50%. Similarly we let the group B be those obtained using the test data labeled by the experiment participant B. Here we use the same name "group A" and "group B" for the values of the F-measure for either of SVM A and SVM B.

We applied the two-sided Mann-Whitney U test [5] with the significance level $\alpha = 0.05$ and observed the test statistic being 20 and 0 for SVM A and SVM B, respectively, concerning the following hypotheses:

- Null hypothesis: there are no difference between the group's A and B.
- Alternative hypothesis: there are difference between groups A and B.

For SVM A, the p-value was obtained as $p = 0.023 \leq \alpha = 0.05$, and the null hypothesis was rejected, so that the values of the F-measure had statistically significant difference between the group's A and B. For SVM B, the p-value was obtained as $p = 0.000157 \leq \alpha = 0.05$, and the null hypothesis was rejected, so that the same conclusion was made. We provide the data used for the testing of SVM A and B in Tables 6 and 7, respectively.

VI. RELATED RESEARCH WORKS AND DISCUSSION

Many approaches were proposed to detect spoilers aiming to detect spoilers in tweets, similarly as in the present study, to detect ones in reviews of books, and so on. Many of them are based on machine learning, while others use keyword matching. Among the detectors based on machine learning, several methods are supervised and others are semi-supervised. We compare the proposed approach with the existing ones from various viewpoints and provide corresponding discussion below.

The simplest approach may be based on the keyword matching method [13, 14] in which spoilers are filtered out by using a set of keywords selected in advance. When using this method, it is necessary to select keywords depending on the topic of the tweets.

Another basic approach is to use the Latent Dirichlet allocation (LDA) [14], which can be applied to classify documents. In [15], the authors suggest exploiting LDA to rank spoilers. However, it is not applicable to short texts, such as tweets.

Several supervised approaches were presented to classify microblogs, such as tweets. The approach presented in [16] was aimed to classify tweets concerning the sentiment, by using deep learning to divide them into two classes: positive and negative. In [17], it was proposed to classify tweets concerning their subjectivity and objectivity. The method suggested in [18] was intended to classify microblogs, such as tweets on Twitter, into two classes: spam and non-spam, by using the knowledge obtained from other media, such as emails. The present research work also belongs to this group of studies: classifying microblogs, such as tweets. The main idea of the present study is to use semi-supervised learning and not supervised one. This approach was not thoroughly investigated, except in work [1], as described below in detail.

Jeon et al. [1] suggested a semi-supervised approach to detect spoilers, as mentioned in Section I. Similarly to the approach proposed in the present study, and they used SVM for classifying tweets into spoilers and non-spoilers.

ISSN: 2414-4592
DOI: 10.20474/jater-5.5.4

TABLE 5

F-MEASURE OF CLASSIFICATION BY SVMS A AND B W. R. T. LABELS ATTACHED BY THE EXPERIMENT PARTICIPANTS A AND B

	F-Measure w.r.t. Labels Attached by the Experiment Participant A	*F*-Measure w.r.t. Labels Attached by the Experiment Participant B
SVM A	0.725	0.714
SVM B	0.701	0.741

TABLE 6

VALUES OF *F*-MEASURE USED FOR TESTING IN SVM A

The Person who Classified the Tweets	*F*-Measure	Order
The experiment participant A	0.728	16
	0.722	10
	0.727	15
	0.724	13
	0.722	11
	0.732	20
	0.729	17
	0.717	8
	0.716	7
	0.730	18
The experiment participant B	0.719	9
	0.710	4
	0.726	14
	0.704	2
	0.724	12
	0.712	5
	0.715	6
	0.709	3
	0.731	19
	0.688	1

TABLE 7

VALUES OF *F*-MEASURE USED FOR TESTING IN SVM B

The Person who Classified the Tweets	*F*-Measure	Order
The experiment participant A	0.702	5.5
	0.702	5.5
	0.700	4
	0.704	8
	0.692	1
	0.704	7
	0.706	9
	0.709	10
	0.699	3
	0.695	2
The experiment participant B	0.745	16
	0.741	15
	0.748	19.5
	0.735	12
	0.745	17.5
	0.738	13
	0.740	14
	0.730	11
	0.748	19.5
	0.745	17.5

The training dataset consisted of tweets, and some of them were labeled by humans, and the others remained unlabeled. Unlike the proposed approach, they utilized the four predefined features of tweets:

• Frequencies of named entities, such as names of soccer players, appearing in the tweets obtained by using the Twitter Named Entity Eecognizer (TNER) [19];

• The objectivity of the tweets [17]: containing the facts rather than emotions;

• Frequencies of the target verbs defined prior to the learning;

• The main tense of tweets.

The above features were inputted into SVM. In contrast, in the present study, we transformed each tweet into a BoW [9] representation and then simplified it into a 64 dimension vector by using latent semantic indexing [10]. This is a

ISSN: 2414-4592
DOI: 10.20474/jater-5.5.4

relatively simple approach compared with that proposed in [1], and yet, the obtained result in terms of the F-measure is comparable to theirs. It should be noted that similarly to the approach proposed in the present paper, they employed datasets composed of the tweets corresponding to the specific topics, such as “Dancing with the Stars” and the final of the 2014 World Cup. They considered several combinations of the above four features, and the best achieved value of the F-measure was 0.7839, while in the conducted experiment corresponding to the present study, the value of the F-measure was 0.741 for SVM B and 0.725 for SVM A, as shown in Section IV. Moreover, as mentioned in I, the most important difference is that in the present study, we demonstrated that the proposed approach could classify tweets depending on the user preferences.

Then, Chang et al. [20] proposed an approach based on deep learning to detect the spoilers that did not require specific features, such as named entities used in other studies, similarly as in [1] by Jeon et al. The system implemented in the study by Chang et al. [20] included the genre encoder and sentence encoder. The latter was used to extract the sentence feature from input sentences in a form of vectors. There, they used a layer to compute the weight referred to as the attention weight corresponding to each word in the input sentence based on the genre. The extracted sentence feature vectors were then inputted into the classifier. In its turn, the proposed system employs the dataset composed of tweets corresponding to a specific topic: professional baseball games. We leave as a future work to develop the detectors that mechanically incorporate genres of tweets.

Shiratori et al. [21] analyzed the spoilers related to soccer games played by the Japan national team and published on Twitter to identify what kind of information spoilers typically presented. They collected the tweets with hashtags, such as daihyo and JPN, asked several college students to label the tweets, and then, analyzed the labeled tweets. Based on the analysis results, they developed a detector based on SVM, similarly to those proposed in the present study and the work by Jeon et al. [1]. They employed BoW representations as an input into the SVM, similarly to that implemented in the present study. The difference with the present study was that they divided tweets by time zones according to a context of the games: winning, tie, and losing. They evaluated the SVM based on the value of F-measure, which was 0.611. In the present study, the values of the F-measure were equal to 0.741 for SVM B and 0.725 for SVM A, which were better than the value 0.611 of the F-measure achieved in the study [21]. Another difference was that their detector was not personalized while the ones proposed in the present paper was.

Ueno et al. [22] developed a detector of spoilers in Japanese. They used a detector proposed by Hijikata et al. [23] to detect spoilers in reviews in English and in Japanese on products with stories, such as books. As a result, they found that detecting spoilers in English achieved better result compared to the task of detecting the ones in Japanese. They observed that this was because the fact that sentences in Japanese often omitted the subjects like “I”, “he”, and “she”. To cope with this problem, they used the vector representation of tweets and arranged the vectors in the chronological order. Thereafter, their detector learned the classification by using a long short-term memory network [24], a recurrent neural network. They evaluated the classifier by the value of the F-measure, which was 0.55. The presently proposed classifier achieves the value of the F-measure much better than theirs, although the target data were different: they considered reviews, and not tweets.

VII. CONCLUSIONS AND FUTURE WORK

In the present study, we proposed a method to prevent displaying spoilers on Twitter based on the personalized SVM and focused on finding the spoilers on professional baseball games of Yokohama DeNA Baystars in Japan. We implemented the semi-supervised approach to detect spoilers and tested its applicability by asking two experiment participants to attach labels to several tweets in the dataset that was then inputted into SVM. We analyzed the obtained values of F-measures by applying the two-sided Mann-Whitney U test with the significance level $\alpha = 0.05$ and observed that the values indicated the statistically significant difference between the two groups of the test data labeled by the two participants, respectively.

In the future, we plan to increase the values of F-measures. Although in the present study, we have employed BOW representations as the feature, we expect that the values can be increased by considering other aspects, such as the burst phenomenon on Twitter, while keeping the property of personalization. Moreover, we plan to develop detectors to incorporate genres of the tweets mechanically.

REFERENCES

[1] S. Jeon, S. Kim, and H. Yu, ``Spoiler detection in tv program tweets,'' *Information Sciences*, vol. 329, pp. 220-235, 2016. doi: https://doi.org/10.1016/j.ins.2015.09.005

ISSN: 2414-4592
DOI: 10.20474/jater-5.5.4

[2] D. Yarowsky, ``Unsupervised word sense disambiguation rivaling supervised methods,'' in *33rd Annual Meeting of the Association for Computational Linguistics,* California, CA, 1995.

[3] C. Cortes and V. Vapnik, ``Support-vector networks,'' *Machine Learning*, vol. 20, no. 3, pp. 273-297, 1995. doi: https://doi.org/10.1007/bf00994018

[4] H. S. Alam, D. Soetraprawata, Bahrudin, T. Haiyunnisa, T. I. Salim, A. Munandar, and D. Setiawan, ``Design of stand-alone irrigation system on strawberry cultivation powered by wind turbine and photovoltaics,'' *International Journal of Technology and Engineering Studies*, vol. 2, no. 5, pp. 154-163, 2016. doi: https://doi.org/10.20469/ijtes.2.40005-5

[5] H. B. Mann and D. R. Whitney, ``On a test of whether one of two random variables is stochastically larger than the other,'' *The Annals of Mathematical Statistics*, vol. 18, no. 1, pp. 50-60, 1947. doi: https://doi.org/10.1214/aoms/1177730491

[6] O. A. Osahenvemwen and O. F. Odiase, ``Effective management of handover process in mobile communication network,'' *Journal of Advances in Technology and Engineering Studies*, vol. 2, no. 6, pp. 176-182, 2016. doi: https://doi.org/10.20474/jater-2.6.1

[7] Git Hub, ``Tweepy,'' n.d. [Online]. Available: https://bit.ly/2LY885U

[8] Git Hub, ``Mocobeta/janome: Japanese morphological analysis engine written in pure python,'' n.d. [Online]. Available: https://bit.ly/3nBCQj4

[9] Z. S. Harris, ``Distributional structure,'' *Word*, vol. 10, no. 2-3, pp. 146-162, 1954. doi: https://doi.org/10.1080/00437956.1954.11659520

[10] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, ``Indexing by latent semantic analysis,'' *Journal of the American Society for Information Science*, vol. 41, no. 6, pp. 391-407, 1990. doi: https://doi.org/10.1002/(sici)1097-4571(199009)41:6<391::aid-asi1>3.0.co;2-9

[11] Pypi, ``Project description,'' n.d. [Online]. Available: https://bit.ly/34z5j1z

[12] Git Hub, ``semisup-learn,'' n.d. [Online]. Available: https://bit.ly/3mMDr0k

[13] J. Golbeck, ``The twitter mute button: a web filtering challenge,'' in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems,* New York, NY, 2012.

[14] S. Nakamura and K. Tanaka, ``Temporal filtering system to reduce the risk of spoiling a user's enjoyment,'' in *Proceedings of the 12th International Conference on Intelligent User Interfaces,* Florida, FL, 2007.

[15] S. Guo and N. Ramakrishnan, ``Finding the storyteller: Automatic spoiler tagging using linguistic cues,'' in *Proceedings of the 23rd International Conference on Computational Linguistics,* Kulalumpur, Malaysia, 2010.

[16] D. Tang, F. Wei, B. Qin, T. Liu, and M. Zhou, ``Coooolll: A deep learning system for twitter sentiment classification,'' in *Proceedings of the 8th International Workshop on Semantic Evaluation,* Bangkok, Thailand, 2014.

[17] J. M. Chenlo and D. E. Losada, ``An empirical study of sentence features for subjectivity and polarity classification,'' *Information Sciences*, vol. 280, no. 5, pp. 275-288, 2014. doi: https://doi.org/10.1016/j.ins.2014.05.009

[18] X. Hu, J. Tang, and H. Liu, ``Leveraging knowledge across media for spammer detection in microblogging,'' in *Proceedings of the 37th international ACM SIGIR Conference on Research & Development in Information Retrieval,* London, UK, 2014.

[19] A. Ritter, S. Clark, and O. Etzioni, ``Named entity recognition in tweets: An experimental study,'' in *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing,* New Dehli, India, 2011.

[20] B. Chang, H. Kim, R. Kim, D. Kim, and J. Kang, ``A deep neural spoiler detection model using a genre-aware attention mechanism,'' in *Pacific-Asia Conference on Knowledge Discovery and Data Mining,*New York, NY, 2018.

[21] Y. Shiratori, Y. Maki, S. Nakamura, and T. Komatsu, ``Detection of football spoilers on twitter,'' in *International Conference on Collaboration Technologies,* London, UK, 2018.

[22] A. Ueno, Y. Kamoda, and T. Takubo, ``A spoiler detection method for Japanese-written reviews of stories,'' *International Journal of Innovative Computing Information and Control*, vol. 15, no. 1, pp. 189-198, 2019. doi: https://doi.org/10.1109/icicic.2006.96

[23] Y. Hijikata, H. Iwai, and S. Nishida, ``Context-based plot detection from online review comments for preventing spoilers,'' in *IEEE/WIC/ACM International Conference on Web Intelligence (WI),* California, CA, 2016.

[24] F. A. Gers, J. Schmidhuber, and F. Cummins, ``Learning to forget: Continual prediction with LSTM,'' *Neural Computation*, vol. 12, no. 10, pp. 2451-2471, 1999. doi: https://doi.org/10.1162/089976600300015015

ISSN: 2414-4592
DOI: 10.20474/jater-5.5.4