Sarcasm Identification on Twitter: A Machine Learning Approach

Aytuğ Onan(✉)

Department of Software Engineering, Faculty of Technology,
Celal Bayar University, 45400 Manisa, Turkey
aytug.onan@cbu.edu.tr

Abstract. In recent years, the remarkable growth in social media and microblogging platforms provide an essential source of information to identify subjective information of people, such as opinions, sentiments and attitudes. Sentiment analysis is the process of identifying subjective information from source materials towards an entity. Much of the social content online contain nonliteral language, such as irony and sarcasm, which may degrade the performance of sentiment classification schemes. In sarcastic text, the expressed text utterances and the intention of the person employing sarcasm can be completely opposite. In this paper, we present a machine learning approach to sarcasm identification. In this scheme, we utilized lexical, pragmatic, dictionary based and part of speech features. We employed two kinds of features to describe lexical information: unigrams and bigrams. In addition, term-frequency, term-presence and TF-IDF based representations are evaluated. To evaluate predictive performance of different representation schemes, Naïve Bayes, support vector machines, logistic regression and k-nearest neighbor classifiers are utilized.

Keywords: Sarcasm identification · Twitter · Machine learning

1 Introduction

Automatic identification of sarcasm is an important problem in natural language processing. With the advances in World-Wide Web (WWW), there is a remarkable growth in social media and microblogging platforms. Hence, a large amount of information is available on the web. This information can serve as an important source to identify subjective information of people, such as opinions, sentiments and attitudes.

Sentiment analysis (also known as opinion mining) is the process of identifying subjective information in source materials. The identification of public sentiment toward policies, products or services can be beneficial to the organizations [1]. Being able to identify subjective information is very important to generate structured knowledge that will serve crucial information to decision support systems and individual decision makers [2]. Much of the social content available on the Web contain nonliteral language, such as irony and sarcasm. For instance, the Internet Argumentation Corpus collected from 4forums.com contains utterances, 12% of which has sarcasm [3]. Sarcastic languages can degrade the predictive performance of sentiment

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_37

classification schemes. In sarcastic text documents, the expressed text utterances and the intention of the person employing sarcasm can be completely opposite.

Automatic identification of sarcasm is in its infancy [4]. One reason is that sarcasm is a hard concept to define. Since sarcasm is an ambiguous concept, it is even difficult for people to precisely identify whether a sentence is sarcastic or not [5]. Another reason is the absence of accurately-labeled naturally occurring utterances labeled as sarcastic that can be used to train supervised learning algorithms [4, 5]. In microblogging platforms, such as Twitter, users can express their opinions, feelings and ideas in short messages called tweets, within 140-character limit. Twitter is a fact growing microblogging platform with over 310 million monthly active users as of June 2016 [6]. Twitter enables users to communicate in a faster mode. Users can use Twitter for several purposes, such as daily chatter, conversation, sharing information and reading breaking news [7]. Recently, Twitter serves as an important source of information for researchers and practitioners, owing to the abundant amount of user-generated text messages on Twitter [8].

In this paper, we present a machine learning approach to identify sarcasm on Twitter messages. We report empirical results of the use of lexical, pragmatic, dictionary based and part of speech features in sarcasm identification. We employed two kinds of features to describe lexical information: unigrams and bigrams. In addition, term-frequency, term-presence and TF-IDF based representations are taken into consideration. In the classification phase, the predictive performance of four supervised learning algorithms (namely Naïve Bayes algorithm, logistic regression algorithm, support vector machines and K-nearest neighbor algorithm) are examined.

The rest of the paper is structured as follows: In Sect. 2, related work on sarcasm identification. Section 3 presents the methodology of the study. In Sect. 4, experimental procedure and empirical results are presented. Finally, Sect. 5 presents the concluding remarks.

2 Related Work

This section briefly reviews the existing works on automated identification of sarcasm. There are many works dedicated to sarcasm and irony in the literature of linguistics, psychology and cognitive science [4, 9–11].

In the domain of text mining, automatic identification sarcasm is considered as a challenging problem [4]. Tepperman et al. [12] examined the predictive performance of prosodic, spectral and contextual features on automatic identification of sarcasm. The empirical analysis indicated that the utilization of contextual features in conjunction with spectral features produces the highest predictive performance in terms of F-measure and classification accuracy. In another study, Kreuz and Gaucci [13] examined the effect of lexical features (such as the use of certain parts of speech tags, punctuation marks, presence of interjections) on automated identification of sarcastic language. Davidov et al. [14] presented a semi-supervised classification scheme to identify sarcastic and non-sarcastic utterances from Twitter messages and product reviews. In another study, Gonzalez-Ibanez et al. [4] presented a machine learning approach to sarcasm identification. In this scheme, unigrams, the presence of dictionary-based lexical and pragmatic factors and the frequency of dictionary-based lexical and pragmatic factors were considered as features and support vector machines and logistic regression algorithms were taken as the supervised learning algorithms. In another study, Veale and Hao [15] presented a rule-based scheme to identify whether a given simile is sarcastic or not. In this scheme, Google search was employed to identify how likely a simile is. In another study, Riloff et al. [16] presented an iterative algorithm that can automatically learn phrases corresponding to positive sentiments and negative situations. The process initiates with a single seed word and a large set of sarcastic tweets. In this scheme, discriminative phrases are learned and the algorithm utilizes the learned list of sentiment and phrases in identification of sarcasm on newer tweets. In another study, Liebrecht et al. [17] examined the use of intensifiers and exclamations in identifying sarcasm on Twitter. They hypothesized that explicit markers (such as hashtags) are digital equivalents of nonverbal expressions indicating sarcasm in live interactions. More recently, Rajadesingan et al. [18] incorporated author-specific contextual information into account in the sarcasm identification. In this scheme, contextual features (such as user's familiarity with twitter, language and sarcasm) are examined. In another study, Bamman and Smith [19] examined the use of extra-linguistic contextual information on sarcasm identification. The empirical analysis indicated that the use of contextual utterance on Twitter, such as properties of author, the audience and immediate communicative environment, enhances the predictive performance of machine learning based classification schemes.

3 The Methodology

This section presents the data set collection, feature engineering utilized to represent sarcasm dataset and classification algorithms utilized in the empirical analysis.

3.1 Dataset Collection

In the dataset collection, we adopted the framework presented in [4]. To build the sarcasm dataset with sarcastic, positive and negative tweets, self-annotated tweets by Twitter users are utilized. Twitter messages with hash tags of "#sarcasm" or "sarcastic" are regarded as sarcastic tweets, whereas hash tags related to positive sentiments are regarded as positive tweets and hash tags related to negative sentiments are regarded as negative sentiments. We used Twitter API to collect tweets. In this way, we have a collection of 5000 sarcastic, 5000 positive and 5000 negative tweets. In order to preprocess the dataset, special characters, such as "@" and "#" are removed from the dataset. In addition, words related to class hash-tags are removed.

3.2 Feature Engineering

In this section, we examine the different feature engineering schemes on the identification of sarcastic utterances. In this scheme, lexical, pragmatic, dictionary-based, part of speech based features are utilized.

Lexical features: We used two kinds of lexical features to represent tweets, namely, unigrams and bigrams are considered. In the vector space model, the frequency of features are taken into consideration to represent text documents. For a particular word *t,* term frequency of *t* in document *d* is defined as *TF*(*t, d*). In addition to term frequency, term presence *TP*(*t, d*) may also be considered to represent features. In this scheme, presence or absence of a word is utilized such that each word *t* is represented by 1 if it is present on a given document *d* and 0, otherwise. Term scoring schemes (such as TF-IDF, term frequency-inverse document frequency) can also be employed to evaluate the importance of a particular word on a given document collection or corpus. Hence, we have utilized term frequency (*TF*), term presence (*TP*) and term frequency-inverse document frequency (*TF-IDF*) to represent tweets.

Pragmatic features: We used three pragmatic features to represent tweets. First, the presence of positive emoticons (such as smileys) is regarded as binary features. Second, the presence of negative emoticons (such as frowning faces) is regarded as binary features. In addition, the presence of words in the interjections list is regarded as a binary feature.

Dictionary-based features: We used NRC word-emotion association lexicon (also known as EmoLex) to derive the dictionary-based features [20, 21]. EmoLex lexicon consists of a list of English words and their associations with eight basic emotions (such as anger, fear, anticipation, trust, surprise, sadness, joy and disgust) and two sentiments (negative and positive). The annotations were manually handled by crowdsourcing on Mechanical Turk. The lexicon contains 14,182 unigrams and approximately 25,000 senses.

Part-of-speech based features: We used the number of words representing positive emoticons and the number of words representing negative emoticons as features. In this scheme, the number of positive emotions, the number of negative emotions and total number of emotions in each tweet are regarded as features.

3.3 Classification Algorithms

In the classification, the predictive performance of four supervised learning algorithms are evaluated. This section briefly explains the algorithms employed in empirical analysis.

Naïve Bayes algorithm (NB) is a probabilistic classification algorithm, which is based on Bayes' theorem [22]. Naïve Bayes algorithm has a simple structure, owing to its assumption of conditional independence. In this way, the required computations are simplified, while obtaining promising predictive performance comparable to other conventional supervised learning algorithms, such as decision trees and artificial neural networks.

Support vector machines (SVM) are supervised learning algorithms that can be employed to solve classification and regression problems. SVM can effectively classify both linear and non-linear data [23]. In SVM, a non-linear matching is employed to transform the original dataset into a higher dimensional hyper-plane. This hyper-plane is used to identify optimal decision boundary that partitions the data into the appropriate classes.

Logistic regression (LR) is a linear classification algorithm, which models the probability of events' occurrence as a linear function of a set of predictor variables [24]. In logistic regression, the decision boundaries are determined based on a linear function of the features. LR aims to optimize the likelihood function to identify class labels for documents. The parameters of LR is chosen so that the conditional likelihood is maximized [25].

K-nearest neighbor algorithm (KNN) is an instance-based classification algorithm that can be employed for classification and regression problems. In KNN, the class label for a particular instance is identified based on the *k*-nearest training instances of a particular instance. The majority voting is employed to combine the predictions of the neighbors of an instance. In this scheme, each instance is assigned to the majority vote of its neighbors, namely, the most common class among its *k*-nearest neighbors [26].

4 Experimental Analysis

This section presents evaluation metrics, experimental procedure and experimental results of empirical analysis.

4.1 Evaluation Measures

In order to evaluate the performance of classification algorithms, two different evaluation measures, namely, classification accuracy and F-measure.

Classification accuracy (ACC) is the proportion of true positives and true negatives obtained by the classification algorithm over the total number of instances as given by Eq. 1:

$$ACC = \frac{TN + TP}{TP + FP + FN + TN} \tag{1}$$

where TN denotes number of true negatives, TP denotes number of true positives, FP denotes number of false positives and FN denotes number of false negatives.

Precision (PRE) is the proportion of the true positives against the true positives and false positives as given by Eq. 2:

$$PRE = \frac{TP}{TP + FP} \tag{2}$$

Recall (REC) is the proportion of the true positives against the true positives and false negatives as given by Eq. 3:

$$REC = \frac{TP}{TP + FN} \tag{3}$$

F-measure takes values between 0 and 1. It is the harmonic mean of precision and recall as determined by Eq. 4:

$$F - measure = \frac{2 * PRE * REC}{PRE + REC} \tag{4}$$

4.2 Experimental Procedure

In the experimental analysis, 10-fold cross validation method is employed. In this scheme, the original dataset is randomly divided into ten mutually exclusive folds. Training and testing process is repeated ten times and each part is tested and trained ten times and the average results for 10-fold are reported. The experimental analysis is performed with the machine learning toolkit WEKA (Waikato Environment for Knowledge Analysis) version 3.9, which is an open-source platform that contains many machine learning algorithms implemented in JAVA. In the empirical analysis, we have performed a three-way comparison of sarcastic (S), positive (P) and negative (N) messages (denoted by PNS). In addition, we have performed three two-way comparisons of sarcastic (S), positive (P) and negative messages (N): namely, negative vs sarcasm classifier (NS), positive vs sarcasm classifier (PS) and non-sarcastic and sarcastic classifier (NSS) are also evaluated.

4.3 Experimental Results

In Tables 1 and 2, classification accuracies and F-measure values obtained by the compared algorithms on sarcasm identification are presented. As it can be observed from the experimental results presented in Tables 1 and 2, the highest predictive performance is obtained from three-way and two-way classification analysis, when term-presence and unigram features are utilized to represent text documents. Compared to bigram features, unigram features generally yield more promising results in terms of classification accuracies. In addition, term presence based representation yields more promising results compared to two other schemes, namely term-frequency based representation and TF-IDF based weighting scheme. Regarding the predictive performance of supervised machine learning algorithms, support vector machines and logistic regression classifiers outperform the other supervised learning algorithms. For negative vs sarcastic classifier (NS), three-way comparison of sarcastic, positive and negative messages (PNS), positive vs sarcasm classifier (PS), the highest classification accuracies are obtained by logistic regression algorithm. For non-sarcastic and sarcastic classifier (NSS), support vector machines yield the highest classification accuracy (89.15%).

Table 1. Classification accuracies obtained by supervised learning algorithms

	Representation	NB	SVM	LR	KNN
NS	TF, unigram	71.55	79.57	80.23	69.63
	TP, unigram	**71.89**	**81.2**	**81.83**	**70.15**
	TF-IDF, unigram	71.82	79.66	80.53	69.97
	TF, bigram	69.07	71.49	72.45	67.71
	TP, bigram	69.54	73.24	73.93	68.36
	TF-IDF, bigram	69.38	71.64	72.75	68.19
PNS	TF, unigram	72.92	73.77	77.64	72.88
	TP, unigram	**73.82**	**77.98**	**78.49**	**73.46**
	TF-IDF, unigram	73.72	74.55	78.38	72.94
	TF, bigram	71.61	72.65	73.36	71.95
	TP, bigram	72.20	73.73	74.39	72.47
	TF-IDF, bigram	71.94	72.69	73.33	72.30
PS	TF, unigram	72.09	78.25	79.29	69.38
	TP, unigram	**73.40**	**80.6**	**81.13**	**69.90**
	TF-IDF, unigram	72.27	78.35	79.62	69.80
	TF, bigram	69.25	71.15	72.24	67.37
	TP, bigram	72.03	73.01	73.47	67.84
	TF-IDF, bigram	69.96	71.33	72.59	67.76
NSS	TF, unigram	72.94	86.38	86.38	86.00
	TP, unigram	**77.14**	**89.15**	**88.87**	**87.12**
	TF-IDF, unigram	75.78	86.71	87.51	86.14
	TF, bigram	74.00	84.05	83.14	84.48
	TP, bigram	75.13	85.67	86.16	86.24
	TF-IDF, bigram	74.21	84.29	85.72	85.31

In Fig. 1, the comparisons of different representation schemes and supervised learning algorithms are provided. In Table 2, F-measure values obtained from the supervised learning algorithms on sarcasm identification are presented. Similar to the empirical results presented in Table 1, the best F-measure results are obtained by term presence and unigram based feature representation. In addition, support vector machines and logistic regression algorithms generally outperform the other supervised learning algorithms in terms of F-measure values. The results presented in empirical analysis indicate that lexical, pragmatic, dictionary-based and part-of speech based features utilized to represent tweets can yield promising results on sarcasm identification. The identification of an appropriate feature set is an important issue in developing robust machine learning based classification schemes. Hence, the experimental results presented in this section may be further enhanced with the use of contextual features and more other feature engineering schemes.

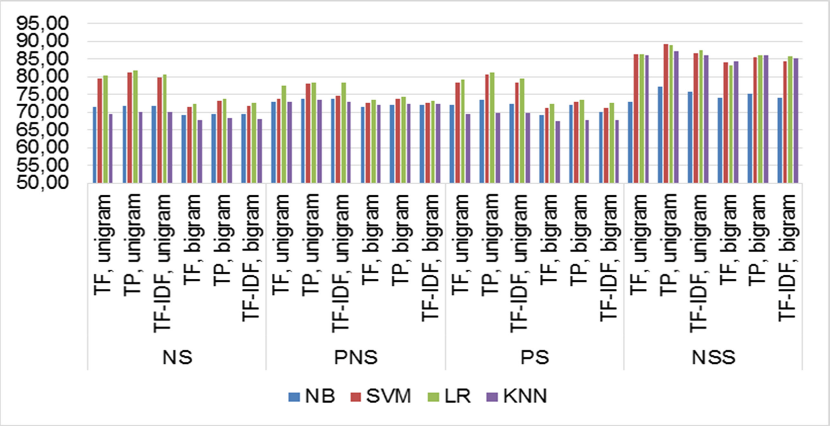

Fig. 1. Average classification rates for algorithms

Table 2. F-measure values by supervised learning algorithms

	Representation	NB	SVM	LR	KNN
NS	TF, unigram	0.81	0.85	0.84	0.75
	TP, unigram	**0.85**	**0.87**	**0.85**	**0.78**
	TF-IDF, unigram	0.84	0.85	0.84	0.77
	TF, bigram	0.77	0.74	0.74	0.70
	TP, bigram	0.80	0.76	0.76	0.74
	TF-IDF, bigram	0.80	0.75	0.74	0.74
PNS	TF, unigram	0.80	0.80	0.79	0.73
	TP, unigram	**0.83**	**0.81**	**0.84**	**0.74**
	TF-IDF, unigram	0.81	0.80	0.80	0.74
	TF, bigram	0.72	0.78	0.78	0.65
	TP, bigram	0.80	0.80	0.78	0.71
	TF-IDF, bigram	0.77	0.79	0.78	0.67
PS	TF, unigram	0.71	0.74	0.73	0.59
	TP, unigram	**0.79**	**0.76**	**0.83**	**0.66**
	TF-IDF, unigram	0.78	0.74	0.73	0.63
	TF, bigram	0.64	0.64	0.63	0.55
	TP, bigram	0.67	0.66	0.64	0.58
	TF-IDF, bigram	0.66	0.64	0.63	0.56
NSS	TF, unigram	0.80	0.86	0.85	0.81
	TP, unigram	**0.86**	**0.89**	**0.86**	**0.86**
	TF-IDF, unigram	0.84	0.87	0.84	0.84
	TF, bigram	0.78	0.85	0.83	0.74
	TP, bigram	0.80	0.86	0.85	0.77
	TF-IDF, bigram	0.78	0.85	0.84	0.76

5 Conclusion

With the advances in information and communication technologies, there is a remarkable growth in social media and microblogging platforms. Microblogging platforms can serve as an important source of information for identifying subjective information of people, such as opinions, attitudes and sentiments. Automatic identification of sarcasm is a challenging problem in natural language processing. In this paper, we have presented a machine learning based approach to identify sarcasm on Twitter messages. We have examined the predictive performance of different representation schemes, such as term-frequency, term-presence and TF-IDF based representations. In addition, two kinds of features (namely, unigrams and bigrams) are utilized to describe lexical information. In the classification phase, the predictive performance of four supervised learning algorithms (namely Naïve Bayes algorithm, logistic regression algorithm, support vector machines and K-nearest neighbor algorithm) are examined. Regarding the empirical results, the highest predictive performance (89.15%) is obtained by term-presence and unigram features, when support vector machines are utilized as the supervised learning algorithm.

References

1. Wang, G., Sun, J., Ma, J., Xu, K., Gu, J.: Sentiment classification: the contribution of ensemble learning. Decis. Support Syst. **57**, 77–93 (2014)
2. Fersini, E., Messina, E., Pozzi, F.A.: Sentiment analysis: Bayesian ensemble learning. Decis. Support Syst. **68**, 26–38 (2014)
3. Walker, M.A., Tree, J.E.F., Anand, P., Abbott, R., King, J.: A corpus for research on deliberation and debate. In: Proceedings of Language Resources and Evaluation Conference, pp. 812–817. ACL, New York (2012)
4. Gonzalez-Ibanez, R., Muresan, S., Wacholder, N.: Identifying sarcasm in Twitter: a closer look. In: Proceedings of the 49th Annual Meeting of the Association for Computation Linguistics, pp. 581–586. ACL, New York (2011)
5. Muresan, S., Gonzalez-Ibanez, R., Ghosh, D., Wacholder, N.: Identification of nonliteral language in social media: a case study on sarcasm. J. Assoc. Inf. Sci. Technol. (2016). doi:10.1002/asi.23624
6. About.twitter.com: Company | About. (2016). https://about.twitter.com/company. Accessed 15 Jan 2017
7. Java, A., Song, X., Finin, T., Tseng, B.: Why we Twitter: understanding microblogging usage and communities. In: Proceedings of the 9th WebKDD Conference, pp. 56–65. ACM, New York (2007)
8. Onan, A.: A machine learning approach to identify geo-location of Twitter users. In: Proceedings of the Second International Conference on Internet of Things, Data and Cloud Computing (2017)
9. Gibbs, R.: On the psycholinguistic of sarcasm. J. Exp. Psychol. **105**, 3–15 (1986)
10. Gibbs, R., Colston, H.L.: Irony in Language and Thought. Taylor and Francis, New York (2007)
11. Utsumi, A.: Verbal irony as implicit display of ironic environment: distinguishing ironic utterances from nonirony. J. Pragm. **32**(12), 1777–1806 (2000)

12. Tepperman, J., Traum, D.R., Narayanan, S.: “yeah right”: sarcasm recognition for spoken dialogue systems. In: Proceedings of the Ninth International Conference on Spoken Language Processing, pp. 1–4. Carnegie Mellon University, Pittsburgh (2006)
13. Kreuz, R.J., Gaucci, G.M.: Lexical influences on the perception of sarcasm. In: Proceedings of the Workshop on Computational Approaches to Figurative Language, pp. 1–4. ACM, New York (2007)
14. Davidov, D., Tsur, O., Rappoport, A.: Semi-supervised recognition of sarcastic sentences in Twitter and Amazon. In: Proceedings of the Fourteenth Conference on Computational Natural Language Engineering, pp. 107–116. ACM, New York (2010)
15. Veale, T., Hao, Y.: Detecting ironic intent in creative comparisons. In: ECAI, pp. 765–770. IOS Press, Amsterdam (2010)
16. Riloff, E., Qadir, A., Surve, P., De Silva, L., Gilbert, N., Huang, R.: Sarcasm as contrast between a positive sentiment and negative situation. In: EMNLP, pp. 704–714. ACM, New York (2013)
17. Liebrecht, C.C., Kunneman, F.A., van Den Bosch, A.P.J.: The perfect solution for detecting sarcasm in Tweets# not. In: Proceedings of the 4th Workshop on Computational Approach to Subjectivity, Sentiment and Social Media Analysis, pp. 29–37. ACL, New York (2013)
18. Rajadesingan, A., Zafarani, R., Liu, H.: Sarcasm detection on Twitter: a behavioral modeling approach. In: Proceedings of the Eight ACM International Conference on Web Search and Data Mining, pp. 97–106. ACM, New York (2015)
19. Bamman, D., Smith, N.A.: Contextualized sarcasm detection on Twitter. In: Ninth International AAAI Conference on Web and Social Media, pp. 574–577. AAAI Press, New York (2015)
20. Mohammad, S., Turney, P.: Crowdsourcing a word-emotion association lexicon. Comput. Intell. **29**(3), 435–465 (2013)
21. Mohammad, S., Turney, P.: Emotions evoked by common words and phrases: using mechanical turk to create an emotion lexicon. In: Proceedings of the NAACL-HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, pp. 26–35. ACL, New York (2010)
22. John, G.H., Langley, P.: Estimating continuous distributions in Bayesian classifiers. In: Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, pp. 338–345. Morgan Kaufmann, San Francisco (1995)
23. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York (1995)
24. Kantardzic, M.: Data Mining: Concepts, Models, Methods and Algorithms. Wiley, New York (2011)
25. Aggarwal, C.C., Zhai, C.X.: A survey of text classification algorithms. In: Aggarwal, C.C., Zhai, C.X. (eds.) Data Mining and Knowledge Discovery Handbook, pp. 77–128. Springer, Berlin (2012)
26. Han, J., Kamber, M., Pei, J.: Data Mining: Concepts and Techniques. Morgan Kaufmann, San Francisco (2011)