SRTTU

Journal of Electrical and Computer Engineering Innovations
(JECEI)
Journal homepage: http://www.jecei.sru.ac.ir

JECEI

Research paper

NSE-PSO: Toward an Effective Model Using Optimization Algorithm and Sampling Methods for Text Classification

R. Asgarnezhad[1], S.A. Monadjemi[2,*], M. Soltanaghaei[1]

[1]*Department of Computer Engineering, Isfahan (Khorasgan) Branch, Islamic Azad University, Isfahan, Iran*

[2]*Faculty of Computer Engineering, University of Isfahan, Isfahan, Iran, and Senior Lecturer, School of continuing and lifelong education, National University of Singapore, Singapore, 119077.*

Article Info

Article History:
Received 7 September 2019
Reviewed 14 December 2019
Revised 25 February 2020
Accepted 27 May 2020

Keywords:
Text classification
Sampling technique
Feature selection
Optimization algorithm
Twitter

*Corresponding Author's Email Address:
monadjemi@eng.ui.ac.ir

Abstract

Background and Objectives: With the extensive web applications, review sentiment classification has attracted increasing interest among text mining works. Traditional approaches did not indicate multiple relationships connecting words while emphasizing the preprocessing phase and data reduction techniques, making a huge performance difference in classification.

Methods: This study suggests a model as an efficient model for sentiment classification combining preprocessing techniques, sampling methods, feature selection methods, and ensemble supervised classification to increase the classification performance. In the feature selection phase of the proposed model, we applied n-grams, which is a computational method, to optimize the feature selection procedure by extracting features based on the relationships of the words. Then, the best-selected feature through the particle swarm optimization algorithm to optimize the feature selection procedure by iteratively trying to improve feature selection.

Results: In the experimental study, a comprehensive range of comparative experiments conducted to assess the effectiveness of the proposed model using the best in the literature on Twitter datasets. The highest performance of the proposed model obtains 97.33, 92.61, 97.16, and 96.23% in terms of precision, accuracy, recall, and f-measure, respectively.

Conclusion: The proposed model classifies the sentiment of tweets and online reviews through ensemble methods. Besides, two sampling techniques had applied in the preprocessing phase. The results confirmed the superiority of the proposed model over state-of-the-art systems.

Introduction

Regarding the explosion of information on the Internet, it is difficult to make decisions based on reviews, tweets, etc. People purchase products on the Internet and give their reviews about them every second. These reviews affect the financial statements in companies noticeably [1] [2] [3] [4] [5] [6] [7] [8] [9]. The main problem is that reviews are in natural language. There is a big gap between reviews in natural language and applications that use structured data. Sentiment Classification is a key tool in this field.

The Sentiment Classification problem is an attractive field in text mining. This field extracts the reviews from the unstructured data on the Internet to organize reviews into two or three classes [10] [11]. Twitter-Sanders-Apple (TSA) generated by Sanders Analytics.

Identifying challenges can be considered in Twitter Sentiment Classification consist of classification accuracy, data sparsity, neutral tweets, and linguistic representational; also, tweets are concise. These problems increase the review classification error. The reason behind that problem is the lack of beneficial relationships between words and sampling techniques applied. Twitter Sentiment Classification is different from other domains in Sentiment Analysis. Almost all movie reviews tend predominantly to be positive or negative; also, no sentiment likely belongs to a feature in a sentence of reviews. With the growing amount of reviews, the effect of review quality on the preprocessing techniques becomes undeniable.

Preprocessing has a principal role in Sentiment Classification. It showed that traditional approaches could not provide enough information for Natural Language Processing analyses. Traditional approaches will add unnecessary complexity; in contrast, words are well indicators for sentiment polarity detection. Traditional BOW did not record multiple relationships between words; hence, we add n-grams to the Bag of Word approach to extract features based on the relationships of the words. Many works combined Machine Learning (ML) algorithms with n-grams. The significant results which were more optimal rather than base classifiers achieved [12] [13]. Specifically, we use n-gram features and sampling techniques in preprocessing steps.

Sentiment Classification approaches can be divided into Lexicon-based, ML [14] [15], and Hybrid approaches. ML aims to optimize an algorithm to increase the system performance using examples and experiences in the past. The ML approaches exist based on three methods like supervised, unsupervised, and semi-supervised. The unavailable labeled dataset is a significant drawback for the supervised methods because they obtain the words with a certain domain. Two critical stages in this context are feature and classifier selection for determining the performance of classification. It has revealed that the ML algorithms like Naive Bayes (NB), support vector machine (SVM) [16], and maximum entropy utilized successfully in many types of research. The current author applied the supervised approaches in conjunction with ensemble methods as different alternatives herein. The acquisition of the domain for words relating to a domain corpus is the main benefit of the lexicon-based approach. A hybrid model combines the services of both them to improve the performance of classification. It is obtaining robust accuracy, and endurance of the two mentioned approaches.

The supervised methods are simple and ordinary; in contrast, ensemble methods like bagging obtain more accurate results. We apply boosting, stacking, and voting as other alternatives in this study. A bagging method assigns equal weights to embedded classifiers, but a boosting method gives a particular importance to each embedded classifier. Their results were good enough; therefore, we add sampling techniques and n-grams to our model to improve more.

In the current study, the effect of different combinations of preprocessing, sampling, Particle Swarm Optimization (PSO), and ensemble techniques investigated on the performance of classification. It is distinct from the existing studies due to employing these different combinations; also, both binary and multi-class classifications are applied. After applying a series of operations in preprocessing phases, we form n-gram features with two sampling techniques to improve the performance of both binary and multi-class classifications. Two weighting mechanisms, term frequency-inverse document frequency (IDF) and term frequency (TF) employed to form the word vector. Two supervised methods, and Ensemble classification method employed. To evaluate the proposed model, TSA datasets considered. However, the Twitter dataset is not available, except Sanders. It seems that weighting feature mechanisms obtained different results on the datasets. As shown in our study, the highest precision obtained through TF mechanism on the TSA2 datasets; whereas, the highest precision was obtained through the TFIDF mechanism on the TSA3 dataset. It also appears that bootstrapping sampling, PSO algorithm achieved higher results on the three datasets. The highest results achieved through our optimized model for the boosting method on the datasets. Experiments showed that our independent-domain approach can improve the classification performance and outperform the existing traditional techniques [4] [17] [18] [19].

We used a sampling technique to emphasize that our model is different and significant in binary and multi-class classifications. Also, concerning more sophisticated methods, this model is simple. Here, our contributions to this research shortened as follows:

- Inspiring Data reduction through sampling technique
- Choosing the best feature by the PSO algorithm
- Improving the performance of the classification model relating ensemble methods

The excess of this article organized as follows: Related work contains a summary of the works. Next, the proposed model presented and evaluated through the experiments explained. Conclusively, the article ended in this article

Related work

Sentiment Classification is attracting considerable attention due to its applications in the new year. Several works proposed to improve the classification performance on the known datasets. Those works differ from each other in the preprocessing, classifiers, and applied datasets. Here, we explain some of these works from 2014 to 2020 on Twitter.

In 2014, Da Silva et al. posed a unique approach for many applications in Sentiment Analysis [3]. It seems that the supervised ML methods obtained high accuracy among other methods; hence, we use the supervised methods. Note that supervised methods have a distinct drawback, an available labeled data. These methods have more computational complexity compared to unsupervised methods. Ensemble Classification methods are other alternatives.

After one year, Tripathi and Naganna in [13] attempted to make a different preprocessing scheme for investigating the behavior of NB and SVM classifiers without sampling technique. They found that n-grams obtained higher results for sentiment analysis and the best accuracy achieved by bigrams.

Tripathy et al. used the combination of TFIDF to produce a digit matrix from the text in 2016 [20]. After one year, Pandey et al. proposed a novel clustering method using a cuckoo algorithm on the TSA dataset in 2017 [4].

In 2018, Trupthi et al. [19] investigated the effective topic modeling methodology Latent Dirichlet Allocation to extract the keywords in a clustering manner. Next, they applied the keywords using the fuzzy c-means approach on the twitter dataset. Vashishtha and Susan in [21] proposed a system to classify the posts in social media through fuzzy rules. They offered a new system, which combines nine fuzzy rules with techniques for Word Disambiguation. They reached 58.9, 59.7, and 68.6% of recall, precision, and f1-score values on the TSA3 datasets, respectively. After one year, Tripathi et al. [23] suggested a novel Map-Reduce based K-means to cluster the large scale data.

The present researchers in 2018 [22] proposed a model named SFT for Twitter Sentiment Classification in 2018. The goal of our model was to investigate the role of weighting feature techniques in Sentiment Classification using supervised methods on the Twitter data set. The applied classifier in the current article based on the SFT model in our previous article. The previous descriptions revealed that no work combined the sampling technique with the ensemble method. Therefore, the current study proposed a novel Sentiment Classification model. In 2020, Abbas et al. [24] offered a classification model with four classifiers, and varying techniques to form a single ensemble classifier. They gained an accuracy of 82.2% on Twitter. Also, Jiang et al. [25] develop a novel Neural Network-based model to conduct the aspect-level Sentiment Classification tasks. Naseem et al. [26] shown a transformer-based method for Sentiment Analysis and applied deep learning and the bidirectional Long Short Term Memory network through omitting noise to heighten the classification performance. They reached an accuracy of 96.2% on airline datasets.

Samad et al. [27] investigated the effect of seven scenarios for text processing on Twitter. Their experiments revealed adverse effects on Sentiment Classification of two common text processing steps: 1) stop word removal; 2) averaging word vectors to represent individual tweets. Word selection from context-driven word embedding showed that only the ten most essential words in Tweets cumulatively produce over 98% of the maximum accuracy.

Sharma and Jain in [28] presented the usage of various ML techniques for collecting tweets and assessing sentiments. After gathering data from twitter, they applied preprocessing and feature extraction stages for the text data. Selection methods based on correlation used and ML classifiers to confirm which classifier gives better results. They obtained an accuracy of 88.2% on the Cambridge Analytica dataset.

The current authors in 2020 [18] proposed a new model based on fuzzy analytic hierarchy on Twitter, namely FAHPBEP. The highest f-measure obtained 90.88 and 90.01% for TSA2 and TSA3, respectively. Also, they in [17] suggested a hybrid model based on ensemble methods on Twitter, namely NSET. The highest f-measure obtained 93.52 and 89.64% for TSA2 and TSA3, respectively. We found that using preprocessing techniques in conjunction with ensemble classification methods may enhance the performance results. We believe that these are unseen combinations of ensemble classification methods. Applying proper alternatives that have not considered in the literature, our obtained results may lead to more accurate performance.

The proposed Model

This article introduces a hybrid model to document-level Sentiment Classification. This approach explores multi-class classifications. In multi-class types, sentiments classified into three classes or more. The model investigates the effects of sampling technique, n-grams, and PSO algorithm using ensemble methods for Sentiment Classification. It seems that the combination of sampling methods, weighing schema, and PSO algorithm can improve the classification performance. The proposed model exploits supervised methods to handle the document-level multi-class Sentiment Classification. A set of operations considered for the preprocessing phase. After studying many classification algorithms [30], we applied the SVM as a baseline classifier. Ensemble classifiers and parameter optimization were two important of our model which used herein. The results indicate that the proposed model outperforms others' works in text classification.

Also, the model gives a higher performance through sampling and PSO algorithms. Our proposed model, namely NSE-PSO, investigates the effects of n-grams, weighting feature mechanisms, sampling techniques, and PSO algorithm using ensemble classification methods, in a stepwise manner. Fig. 1 exposes the phases of our proposed NSE-PSO model.

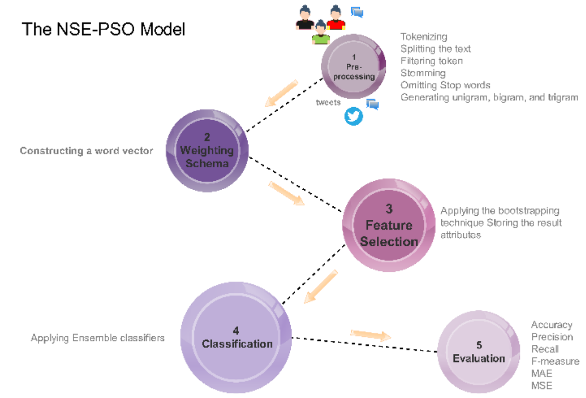

Fig. 1: The phases of the proposed model.

First, the unimportant and worthless characters tokenized and then removed from the text. Next, we employ n-gram features and two weighting feature mechanisms for forming the word vector. It produces to the dataset reduced effectively. Attribute subset selection techniques used to improve classification performance. The PSO algorithm is applied to select the best features. Finally, the classification phase performed by supervised and ensemble classification methods. An extensive range of comparative experiments done on the datasets to investigate the effectiveness of our proposed model. By experiments, the state-of-the-art result obtained. We used the linear SVM as a base classifier in bagging and boosting methods. The comparison among the best results in the literature and our obtained results handled. We employed a type of sampling technique to emphasize that the obtained results are different and significant. In contrast to other sophisticated methods, our hybrid model outperforms good enough in this context.

Here, the pseudo-code of our model expressed as:

Pseudo-code for the NSE-PSO model

Pseudo-code for the preprocessing

```
Input: A dataset;
Output: A classification model for a set of words;
For each document in dataset do
    Tokenizing characters, words, and useless tokens
    Splitting the text into a sequence of tokens
    Filtering tokens based on their length
    Stemming word via Porter Algorithm
    Omitting Stop words based on the stop word list
    Generating unigram, bigram, and trigram
End.
```

Pseudo-code for the weighting schema

```
Input: A set of words;
Output: A set of word vectors;
For each set of words in Input do
    Constructing a word vector based on TF & IDF schemas
End.
```

Pseudo-code for the sampling techniques

```
Input: A set of word vectors;
Output: A set of word vectors;
For each word vector in Input do
    Applying the sampling techniques and storing the result attributes.
End.
```

Pseudo-code for the Particle Swarm Optimization and Classification

```
Input: Training set;
Output: A composite model and confusion matrix;
Maximum iteration, Population size, inaction weight;
Generate initial population;
For each step in maximum iteration do
    Perform the ensemble model for each set of parameters
(Classifying and calculating the fitness function);
        Update the fitness function;
    Stopping and obtaining the optimal parameters;
```

In the following, the phases of the NSE-PSO model are discussed in detail.

A. Preprocessing and Weighting Mechanism Phases

The preprocessing stage consists of Tokenization, Filter-Token, Stemming, Filtering Stop Word, and N-grams. Two weighting feature mechanisms used to create word vectors. The TF defined as total occurrences of the word *t* in document *d* divided by a total volume of the comments happening in document *d*. The TFIDF defined by Manning et al. in (1) as,

$$TFIDF = (TF) \times Log\left(N / F_t\right) \qquad (1)$$

where, TF is the frequency of word t in document *d*, N is the volume of documents in a group, and Ft is the volume of documents in a collection containing word *t* [29].

B. Sampling and Feature Selection Phases

Here, we describe the sampling techniques and PSO algorithm in our model. These techniques caused to select the best features through data reduction. Stratified and bootstrapping samplings are two of the sampling techniques, which used to attain better performance. In stratified sampling, the folds of the training set stratified. The class distribution for tuples in a fold is similar to the initial data. It enables the algorithm to preserve the distribution of the training set. Bootstrapping sampling creates a bootstrapped sample from the dataset. This type of sampling may not have all unique examples; hence it is different from other sampling techniques. We use both samplings in our model, but bootstrapping sampling with replacement achieves higher performance. We apply to bootstrap sampling in all of the experiments. The obtained results show that bootstrapping sampling gives a higher performance on the text input.

To produce an experimental distribution of sampling, a sampling structure of scores was applied. Sampling with or without replacement is the first portion. The size of the sample is the second portion. When a full sampling with replacement happens, the bootstrap is defined as a procedure herein. This bootstrap applies the dimensions of the sample equal to the size of the original collection. In the second portion, a taxonomy determined by extending the sizes of the possible sample. Here, a distribution of the sampling with dimensions more extensive than the dimension of the original data not be applied. Nevertheless, this event did not consolidate for the samples without replacement. Therefore, this proposal only admits expansion for the bootstrap. The tuple likely selects and adds to the training set again when we choose each tuple of the input set. In each step of iterations, all examples have an equal probability of being selected. When an example chooses, it remains a candidate for further selection and determines again in the next step. It is undeniable that a sample with replacement can have the same examples. Therefore, it used to create a sample that is greater in size than the original one. It appears that bootstrapping performs better than the former one. The idea behind bootstrapping is using the data as an input set for approximating the sampling distribution. It creates an enormous volume of samples named bootstrap samples. The sample outline calculated for each sample of bootstrap [30] [31]. Here, some notations for utility described. Assume a parameter for a population θ is as a target herein. An random sample with size *n* produces the data $(X_1, X_2, ..., X_n)$. Assume θ is a sample created from the dataset. The distribution of θ with large n can be bell-shaped with a center θ and standard deviation $\sigma/\sqrt{n}$ for each sample, where the positive volume depends on two factors like the population, and the type of statistic θ. There exist technical complexity for standard deviation, when θ is median or correlation of sample. Therefore, bootstrap assigns a bypass. Assume θ_B is considered as a quantity for presenting the same statistic, which produces on a bootstrap sample of $(X_1, X_2, ..., X_n)$.

With the limitation of $(n \to \infty)$, the distributions of θ_B were bell-shaped with θ as the center and the corresponding standard deviation $\sigma/\sqrt{n}$. Thus, the distribution $\theta_B - \theta$ will be the distribution $\theta - \theta$. This distribution is named the bootstrap Central Limit Theorem (CLT) [32] [33].

It also observed that with a limiting distribution of the sampling for a mathematical function that does not include population unknowns, bootstrap distribution assigns a better conjecture than the CLT. If the procedure is $(\hat{\theta}_B - \hat{\theta})/SE$, where SE considered as a sample estimation of the standard error of $\hat{\theta}$, the limiting sampling distribution will be standard normal.

Here, $\theta = \mu$ is the population means, $\theta = \overline{X}$ is the sample mean, σ is population standard deviation, and s is sample standard deviation considered, which produced from the original dataset. Also, s_B is the sample standard deviation, which calculated on a bootstrap sample. Next, the sampling distribution of $(\overline{X} - \mu)/SE$, with $SE = \sigma/\sqrt{n}$, will be estimated through the bootstrap distribution of $(\overline{X_B} - \overline{X})/SE$, where $\overline{X_B}$ is bootstrap sample means, and $SE = s/\sqrt{n}$. Likewise, the sampling distribution of $(\overline{X} - \sigma)/SE$, where $SE = s/\sqrt{n}$, will be assessed through the bootstrap distribution of $(\overline{X}_B - \overline{X})/SE_B$, where $SE = s/\sqrt{n}$. Here, the description of the approximating standard error of sample evaluation for utility is of concern. We assume that the information investigated regarding the population parameter of θ, where $\overline{\theta}$ is a sample estimator of θ based on a stochastic sample has size *n*. To estimate the standard error for $\hat{\theta}$, a bootstrap approach is of concern: calculate $(\theta_1^*, \theta_2^*, ..., \theta_N^*)$, through the equivalent relation for $\hat{\theta}$, exactly with N numbers of different bootstrap samples. A primary recommendation for the size N could be $N = n^2$, unless n^2 be too large. In that case, it could be reduced to an acceptable size, say $n \log_e^n$. So, $SE_B(\hat{\theta})$ defined as (2),

$$SE_B(\hat{\theta}) = \sqrt{\frac{\sum_{i=1}^{N} (\theta_i^* - \hat{\theta})^2}{N}} \tag{2}$$

It revealed that more instances could exploit more useful information about the dataset. Therefore, it may consider a novel example in the dataset, which is noticeable for classification. That is why bootstrapping sampling has become a useful tool in our model.

PSO is a global technique for optimization problems, proposed by Kennedy et al. [34]. A group of particles represented by x_{ij} search of the solution space to obtain the best solution. Each particle has a place, velocity, and memory to preserve its best position. The rate represented by v_i. This technique is well-known for ease of implementation, convergence speed, and few parameters to adjust, whereas a particle may converge on a suboptimal solution. Each candidate's answer could interpret as a particle with a place in the state space. With particle movement in that space, the optimal solutions emerged. Within a change, each particle refreshes its location and speed according to its neighbors. The best previous place of the particle registered as the individual best p-best, and the best location captured through the population of g-best. Cognitive and social scaling parameters are known as c_1, and c_2. So, the obtained optimal answers through renewing the speed and place of each particle are of concern, (3):

$$V_{ij}^{r+1} = V_{ij}^{r+1} + C_1 rand_1 (p-best_{ij} - X_{ij}^{r}) + C_2 rand_2 (g-best_{ij} - X_{ij}^{r})$$
$$X_{ij}^{r+1} = X_{ij}^{r} + V_{ij}^{r+1} \tag{3}$$

Ultimately, the algorithm stopped by a predefined criterion like a proper fitness amount or a maximum number of iterations. Here, the PSO chooses particles randomly to explore the optimal particle. Per particle is represented as an m-dimensional point/node. The AdaBoost classifier is applied to evaluate effectiveness by the cross-validation technique. PSO investigates the determination of possible subsets to achieve the most significant accuracy. When the efficiency of AdaBoost converges, the iterations end. As mentioned before, PSO configured with population size, inaction weight, and maximum iteration to generate the initial population. The most useful alternative for the local and global parameters examined by evaluating the fitness function of each particle. Then, the speed and location of each particle updates for the values of the fitness converges. Finally, the global best applied for the training of the AdaBoost.

C. Classification phase

Ensemble classification methods applied to obtain better performance. An ensemble combined weak learners to produce a strong learner. It mainly requires more computation to evaluate the prediction; whereas, a single model requires fewer calculations. However, ensemble techniques are a tool for improving the poor performance of base learning algorithms. When the ensemble method trained, it determines a hypothesis. This hypothesis did not contain only the models used for its training; hence, ensembles are more flexible in their functions. However, it can end in over-fitting over them. These methods often tend to yield better results when used in diverse models. Almost all ensemble methods use a diversity of the models to improve the poor performance for each single learners. Among the mentioned classifiers, AdaBoost outperforms the others better. It found that ensemble methods, especially AdaBoost, employed to increase the precision and accuracy through combining a series of individual classifiers. SVM is an ML technique based on the statistical learning concept, which performed well in text classification applications. In the current study, we use a boosting model in conjunction with SVM as base learners. We focus on using sampling and ensemble methods as a base learner for classification. Here, several methods incorporated in the proposed model. In comparison with the literature, our model achieves higher results. That is because the usage of ensemble methods in conjunction with sampling techniques and PSO algorithms gives the best features.

D. Data sets and Evaluation Phase

There are a few available and free resources on Twitter. None of the existing datasets on Twitter are free, except the Sanders dataset. Hence, two datasets of TSA applied for training and testing experiments. The used datasets were generated by Sanders Analytics. Two datasets of the TSA are accessible at http://www.sananalytics.com.

Here, measures for evaluating Sentiment Classification had introduced. P and N are the numbers of positive and negative tuples. TP refers to the positive tuples that have been labeled by the classifier correctly. TN refers to the number of true negatives. FP is the negative tuples that have labeled incorrectly as positive. FN is the positive tuples that have mislabeled as negative. Accuracy is the sum of actual tuples that classified TP and the number of TN relative to the total number of classified instances. The precision state as the percentage of tuples that have labeled as positive and actual. Recall refers to the percentage of tuples that are labeled positive. F-measure combines precision and recall into a single measure [12] [35] [29]. F-measure comes from a weighted harmonic mean of precision and recall. Also, mean absolute error (MAE) and root absolute error (RAE) for error evaluation employed. These measures computed in (4) to (9).

$$Accuracy = \frac{TP+TN}{P+N} \tag{4}$$

$$Precision = \frac{TP}{TP+FP} \tag{5}$$

$$Recall = \frac{TP}{P} \tag{6}$$

$$F-measure = \frac{2PR}{P+R} \tag{7}$$

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} \tag{8}$$

$$RAE = \frac{\sum_{i=1}^{n} |x_i - \overline{x_i}|}{\sum_{i=1}^{n} |y_i - \overline{y_i}|} \tag{9}$$

Results and Discussion

In all experiments, we assume epsilon and the kernel type of the SVM is one and C-SVC, respectively. The remaining parameters for SVM optimized in all experiments. Also, the evaluation model of the kernel NB supposed greedy, with the number of kernels equals 10. During the experiments, the 10-fold cross-validation utilized again, and performance evaluation parameters calculated. In some cases, the random seed used to have different examples. By changing the value of this parameter, we can change the way examples are randomized. Here, a maximum number of iterations for the PSO algorithm considered 100. Implementation of R programming used to conduct the experiments.

Experiment I. Here, we investigate the effect of the PSO algorithm for choosing the best features on the TSA2 dataset. Table 1 shows the obtained results for our model. The highest accuracy, f-measure, and precision values obtained 92.61, 94.49, and 97.33%, respectively. It also seems that the performance of used model using IDF mechanism is higher than that of the TF on the TSA2 dataset. The reason is that each example in this dataset has one or two sentences. It found that the highest performance for the proposed model obtained relating to PSO, IDF, sampling, and bigrams in general. It observed that bootstrapping sampling can improve the performance measure in most cases. This is due to bootstrapping sampling produces a dataset greater than the original dataset. Here, the accuracy and f-measure improved rather than without sampling approximately 10 and 12%, respectively. Fig. 2 presents the highest performance of the experiment I.

Table 1: The obtained results of the proposed method in experiment I

Pre.		P	R	F	A	MAE	RAE
The results after using Sampling and PSO							
IDF	**N=1**	96.50	91.26	90.88	92.19	0.1334	0.2354
	N=2	97.33	91.82	94.49	92.61	0.1365	0.2357
	N=3	96.77	90.48	93.52	91.77	0.1365	0.2312
TF	**N=1**	90.82	86.13	88.41	87.82	0.1312	0.2343
	N=2	95.91	90.81	93.29	91.99	0.1342	0.2312
	N=3	95.49	88.81	92.02	91.37	0.1312	0.2343
No-sample		86.51	79.27	82.73	82.80	0.1256	0.2358

Note: Pre= Preprocessing, P=Precision, R=Recall, F=F-measure, A=Accuracy

Experiment II. We try to investigate the effect of the NSE-PSO on the TSA3 dataset. The AdaBoost method with SVM applied. Table 2 presents the obtained results of this experiment. It revealed that the highest accuracy, precision, recall, and f-measure rates obtained using IDF and PSO. The achieved f-measure of our model were more excellent than 96%; whereas, without sampling were 86.60%. The best-obtained accuracy was 88.75% through the IDF mechanism, unigrams, PSO, and sampling. The highest precision relating PSO, sampling, and trigrams was 95.33%; whereas, the f-measure using sampling, PSO, and bigrams gets through 96.23%. All best performances of the NSE-PSO without PSO achieved applying the TFIDF, bigrams, and unigrams. It appears that the usage of n-grams and PSO increases the results of the usage of sampling.

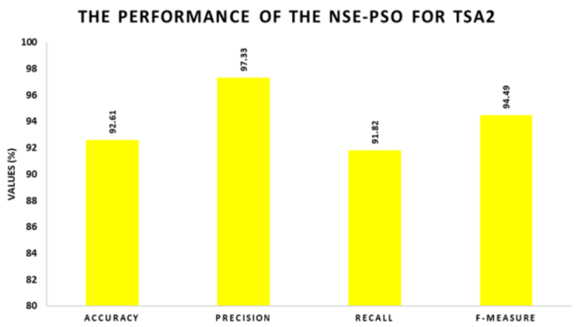

Fig. 2: The comparison between the obtained results using PSO algorithm on the TSA2 dataset.

Table 2: The obtained results of the proposed method in experiment II

Pre.		P	R	F	A	MAE	
The results after using Sampling and PSO							
IDF	**N=1**	95.33	97.16	96.23	88.75	0.1243	0.2225
	N=2	93.76	96.12	94.92	87.74	0.1263	0.2234
	N=3	94.52	94.39	94.45	87.43	0.1254	0.2265
TF	**N=1**	89.51	88.03	88.76	82.58	0.1265	0.2276
	N=2	93.13	87.47	90.21	84.90	0.1222	0.2243
	N=3	93.41	86.91	90.04	84.50	0.1254	0.2243
No-sample		64.66	73.05	68.60	72.90	0.1256	0.2377

Note: Pre= Preprocessing, P=Precision, R=Recall, F=F-measure, A=Accuracy

It also shows that our model on the TSA3 dataset gives higher results using the IDF mechanism. The best general performance achieved by our optimized implementation for the boosting method on the TSA3. The highest f-measure achieved is 96.23%, which belongs to unigrams and PSO. The use of ensembles leads to additional computational costs, but the obtained accuracy is usually worthwhile. The improvement of accuracy is 15% in this experiment rather than the obtained results without sampling. The unnecessary features tokenized and filtered to apply the BOW technique. However, PCA applied as a data preprocessing, but the performance of classification does not improve. Fig. 3 exposes the best results of the NSE-PSO model before and after using sampling.

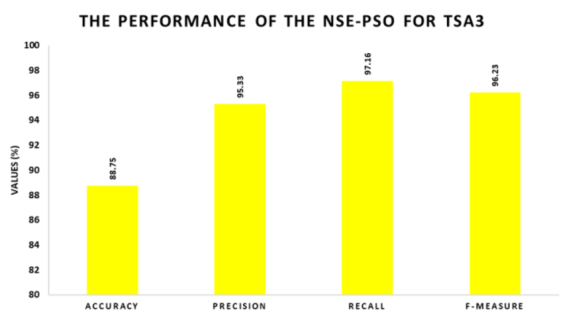

Fig. 3: The comparison between the obtained results using PSO algorithm on the TSA3 dataset.

Experiment III. Here, the best-obtained results from our proposed model and the best results in the literature on the datasets compared. Figs. 4 and 5 show the comparison among the NSE-PSO and three best works in this context.

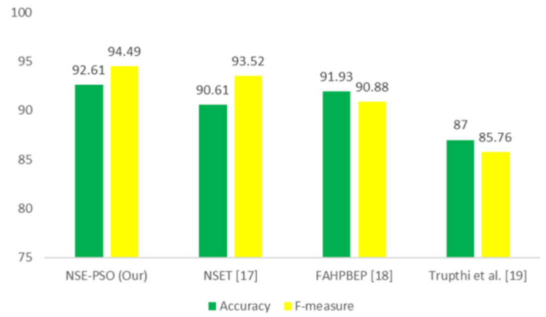

Fig. 4: The comprehensive comparison among our performance of experiment I on the TSA2 dataset.

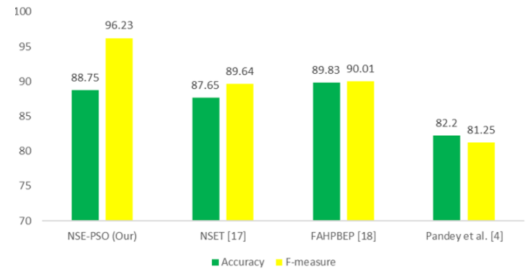

Fig. 5: The comprehensive comparison among our performance of experiment II on the TSA3 dataset.

It appears that bigram features can improve performance. Also, sampling can aim to obtain higher results. The experimental results emphasize that AdaBoost using sampling is substantially more accurate than other techniques in our tests. These results achieved using both TF and IDF mechanisms. It can also found that bigrams can improve the accuracy and f-measure for the two datasets. It seems that we decrease approximately the run time of ensemble methods through sampling. The striking comparison confirmed the advantage of our model over the state-of-the-art systems. It reveals that the proper use of sampling techniques and ensemble classification methods can improve the performance. It also seems that our model with using sampling, bigrams give higher results rather than other methods in the literature on the TSA datasets. According to Fig. 4, the highest accuracy of the proposed model obtained 92.61% on the TSA2 dataset; whereas, Trupthi et al. [19], the NSET in [17], and the FAHPBEP in [18] obtained 87, 90.61, 91.93%, respectively. This is due to that the ensemble method suggested in the NSET used only Adaboost in conjunction with SVM; whereas, we applied sampling and PSO besides boosting to select relevant word in the preprocessing phase. Also, the FAHPBEP and Trupthi et al. used fuzzy approach; whereas, we apply optimization algorithm. The highest f-measure of the proposed model obtained 94.49% on the TSA2 dataset; whereas, Trupthi et al. [19], the NSET in [17], and the FAHPBEP in [18] obtained 85.76, 93.52, 90.88%, respectively. Our obtained f-measure rate was approximately 1% greater than the best in the literature, i.e., 93.52%.

According to Fig. 5, we obtained the accuracy rate using the boosting method of 88.75% on the TSA3 dataset. Whereas Pandey et al. [4] obtained 82.20% of accuracy. They derived features from the k-means and cuckoo search, but we used a PSO algorithm. Despite using the bootstrap model and several classifiers, their framework was not more effective than our approach. It appears that IDF is the best weighting schema for TSA3. The reason for it is the few volume sentences in each example of the dataset. Also, the highest f-measure of our proposed model obtained 96.23% on the TSA3 dataset; whereas, Pandey et al. [4] obtained 81.25%. Notwithstanding applying the k-means model and cuckoo search, their framework was good enough and not more effective. Also importantly and opposition to the other works, higher classification f-measure achieved by our model. The accuracy rate reported of the NSET model in [17] on the TSA3 dataset was 87.65%; whereas, our implementation obtained an accuracy of 88.75%. The improvement of the f-measure rate in our model was approximately 6% greater than the best f-measure rate in the literature. The highest obtained f-measure of our model is 96.23% for the TSA3 dataset; whereas, the accuracy rate of the FAHPBEP in [18] gets through 90.01%. However, our accuracy rate obtained 88.75%, which was approximately 1% lower than the FAHPBEP. It revealed that the f-measure rate can be more significant rate for comparison on the TSA dataset. It revealed that bootstrapping gives the best results on all datasets. The benefit of utilizing ensemble methods is enhancing the classification performance; contrary to time, which it needs to cease the training phase of these methods is a disadvantage. Our concern was constructing an approach that has higher performance compared to the other works on three datasets. No work has been published on Sentiment Classification using the combination of sampling, n-grams, PSO algorithm, and ensemble methods. The proposed model not only applies the ensemble method and sampling but also utilizes a meta-classifier. It seems that sampling and PSO yield better results compared to these existing methods. The obtained results emphasize that AdaBoost using PSO algorithm is substantially more accurate than other applied techniques in our model. Improvement of f-measure calculated to estimate the performance of the present model on the datasets. The improvement measures are of concern in (10):

$$improvement\ of\ f-measure = \\ =(f-measure_{NSE-PSO} - f-measure)/f-measure_{NSE-PSO} \quad (10)$$

For the TSA2 dataset, the improvement of f-measure is 0.97%. For the TSA3 dataset, the progress of f-measure is 6.22%. The benefit of utilizing PSO algorithm is enhancing the classification performance; in contrast, time that needs to cease the training phase of the method is a disadvantage. We used sampling techniques to decrease this time. Our goal was the construction of a model has higher performance compared to the other works. We show that using preprocessing techniques in conjunction with ensemble classification methods may enhance the performance results. The combination of our preprocessing with ensembles and applied optimization algorithm could receive significant improvement than other works. It is the reason that the proposed model outperforms the benchmarks.

Conclusion

People purchase products on the Internet and give their reviews about them every second. These reviews affect the financial statements in companies noticeably. With the explosion of information on the Internet, it is difficult for ordinary people to make decisions about products. Sentiment Classification is a significant field in text mining that can help companies. The proposed model investigates meta-classifiers to increase the performance of the classification for Sentiment Classification on the Twitter datasets. We investigated the effects of the combination of sampling technique, PSO algorithm, and ensemble method on the classification performance. We characterized two weighting mechanisms and n-grams. The goal of our article was the suggestion of the effective model to increase the classification performance for classification tasks. PSO applied as a simple technique for solving optimization problems, which implies well-known ease of implementation and its convergence speed. Sampling as a particular technique is the superior approach; the bootstrapping method used and parameters of SVM optimized to improve the models. Boosting ensemble method in conjunction with SVM employed as base classifiers. The main advantage of the proposed comprehensive model is applying sampling techniques and PSO algorithm in preprocessing steps. We demonstrated the robustness of our model on the datasets. It appears that the IDF mechanism, bigram features, and the combination of them via bootstrapping sampling and PSO reaches the highest performance on the TSA datasets. We conclude that our investigation can be applicable for different social media analyses in the proposition of using ensemble technique, PSO, and bootstrapping the performance for Sentiment Classification. As future work, we are going to study the effect of other optimization algorithms and sampling techniques in this context.

Author Contributions

R. Asgarnezhad designed the experiments, carried out the data analysis, interpreted the results and wrote the manuscript. S. A. Monadjemi corrected the proofing the article. Soltanaghaei supported the article.

Acknowledgment

We thank the editor and all anonymous reviewers.

Conflict of Interest

The authors declare that they have no conflict of interest.

Abbreviations

TFIDF	Term Frequency-Inverse Document Frequency
TF	The frequency of word *t* in document *d*
N	The number of documents
Ft	The number of documents including word *t*
V_{ij}	The velocity of particles
x_{ij}	A group of particles
c_1	Cognitive parameter
c_2	social scaling parameter
MAE	Mean Absolute Error
RAE	Absolute Error

References

[1] E. Kouloumpis, T. Wilson, J.D. Moore, "Twitter sentiment analysis: The good the bad and the omg!," in Proc. Fifth International AAAI conf. on weblogs and social media: 538-541, 2011.

[2] F.H. Khan, S. Bashir, U. Qamar, "TOM: Twitter opinion mining framework using hybrid classification scheme," Decision Support Systems, 57: 245-257, 2014.

[3] N.F. Da Silva, E.R. Hruschka, E.R. Hruschka, "Tweet sentiment analysis with classifier ensembles," Decision Support Systems, 66: 170-179, 2014.

[4] A.C. Pandey, D.S. Rajpoot, M. Saraswat, "Twitter sentiment analysis using hybrid cuckoo search method," Information Processing & Management, 53: 764-779, 2017.

[5] H. Saif, M. Fernández, Y. He, H. Alani, "On stopwords, filtering and data sparsity for sentiment analysis of twitter," in Proc. Ninth International Conf. on Language Resources and Evaluation: 810–817, 2014.

[6] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, B. Qin, "Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification," in Proc. The 52nd Annual Meeting of the Association for Computational Linguistics: 1555-1565, 2014.

[7] B. Besbinar, D. Sarigiannis, P. Smeros, "Tweet Sentiment Classification," Lausanne, 2014.

[8] A. Montejo-Ráez, E. Martínez-Cámara, M. T. Martín-Valdivia, L. A. Ureña-López, "Ranked wordnet graph for sentiment polarity classification in twitter," Computer Speech & Language, 28: 93-107, 2014.

[9] D.-T. Vo, Y. Zhang, "Target-Dependent Twitter Sentiment Classification with Rich Automatic Features," in Proc. IJCAI: 1347-1353, 2015.

[10] A. Go, R. Bhayani, L. Huang, "Twitter sentiment classification using distant supervision," CS224N Project Report, Stanford, 1: 1-6, 2009.

[11] L. Jiang, M. Yu, M. Zhou, X. Liu, T. Zhao, "Target-dependent twitter sentiment classification," in Proceedings of the 49th

Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-1: 151-160, 2011.
[12] A. Tripathy, A. Agrawal, S.K. Rath, "Classification of sentiment reviews using n-gram machine learning approach," Expert Systems with Applications, 57: 117-126, 2016.
[13] A.K. Tripathi, K. Sharma, M. Bala, "Parallel hybrid bbo search method for twitter sentiment analysis of large scale datasets using mapreduce," International Journal of Information Security and Privacy (IJISP), 13: 106-122, 2019.
[14] H. Saif, Y. He, H. Alani, "Alleviating data sparsity for twitter sentiment analysis," in Proc. the 21st International Conference on theWorld Wide Web: 2–9, 2012.
[15] L. Chen, W. Wang, M. Nagarajan, S. Wang, A. P. Sheth, "Extracting Diverse Sentiment Expressions with Target-Dependent Polarity from Twitter," ICWSM, 2: 50-57, 2012.
[16] R. Asgarnezhad, K. Mohebbi, "A Comparative Classification of Approaches and Applications in Opinion Mining," International Academic Journal of Science and Engineering, 2(1): 68-80, 2015.
[17] S. Monadjemi, R. Asgarnezhad, M. Soltanaghaei, "A High-Performance Model based on Ensembles for Twitter Sentiment Classification," Journal of Electrical and Computer Engineering Innovations (JECEI), 8(1): 41-52, 2020.
[18] R. Asgarnezhad, S.A. Monadjemi, M. Soltanaghaei, " FAHPBEP: A fuzzy Analytic Hierarchy Process framework in text classification," accepted in Majlesi Journal of Electrical Engineering, vol. 14, no. 3, 2020.
[19] A.K. Tripathi, K. Sharma, M. Bala, "Parallel hybrid bbo search method for twitter sentiment analysis of large scale datasets using mapreduce," International Journal of Information Security and Privacy (IJISP), 13: 106-122, 2019.
[20] S.H. Seyyedi, B. Minaei-Bidgoli, "Enhancing effectiveness of dimension reduction in text classification," International Journal on Artificial Intelligence Tools, 26(3): 1-21, 2017.
[21] S. Vashishtha, S. Susan, "Fuzzy rule based unsupervised sentiment analysis from social media posts," Expert Systems with Applications, 138: 1-15, 2019.
[22] R. Asgarnezhad, S.A. Monadjemi, M. Soltanaghaei, A. Bagheri, "SFT: A model for sentiment classification using supervised methods in Twitter," Journal of Theoretical & Applied Information Technology, 96(8): 2242-2251, 2018.
[23] A.K. Tripathi, K. Sharma, M. Bala, "Parallel hybrid bbo search method for twitter sentiment analysis of large scale datasets using mapreduce," International Journal of Information Security and Privacy (IJISP), 13: 106-122, 2019.
[24] A.K. Abbas, A. K. Salih, H. A. Hussein, Q.M. Hussein, S.A. Abdulwahhab, "Twitter Sentiment Analysis Using an Ensemble Majority Vote Classifier," Journal of Southwest Jiaotong University, 55: 1-7, 2020.
[25] N. Jiang, F. Tian, J. Li, X. Yuan, J. Zheng, "MAN: mutual attention neural networks model for aspect-level sentiment classification in SIoT," IEEE Internet of Things Journal, 7: 2901-2913, 2020.
[26] U. Naseem, I. Razzak, K. Musial, M. Imran, "Transformer based Deep Intelligent Contextual Embedding for Twitter sentiment analysis," Future Generation Computer Systems: 1-35, 2020.
[27] M.D. Samad, N.D. Khounviengxay, M.A. Witherow, "Effect of Text Processing Steps on Twitter Sentiment Classification using Word Embedding," arXiv preprint arXiv:2007.13027: 1-14, 2020.
[28] S. Sharma, A. Jain, "An Empirical Evaluation of Correlation Based Feature Selection for Tweet Sentiment Classification," in Proc. Advances in Cybernetics, Cognition, and Machine Learning for Communication Technologies, ed: Springer: 199-208, 2020.
[29] C.D. Manning, P. Raghavan, H. Schütze, Introduction to information retrieval vol. 1: Cambridge university press Cambridge, 2008.
[30] J.Han, M. Kamber. Data mining: concepts and techniques. Morgan Kaufmann Publishers–An Imprint of Elsevier, 500: 105-150, 2006.
[31] T.C. Hesterberg, "What teachers should know about the bootstrap: Resampling in the undergraduate statistics curriculum," The American Statistician, 69: 371-386, 2015.
[32] M.R. Chernick, W. González-Manteiga, R.M. Crujeiras, E.B. Barrios, Bootstrap methods. Springer, 2011.
[33] J.S. Haukoos, R.J. Lewis, "Advanced statistics: bootstrapping confidence intervals for statistics with “difficult” distributions," Academic emergency medicine, 12: 360-365, 2005.
[34] R. C. Eberhart, Y. Shi, J. Kennedy, Swarm intelligence: Elsevier, 2001.
[35] E. Fersini, A. Messina, F.A. Pozzi, "Sentiment Analysis: Bayesian Ensemble Learning," Decision Support Systems, 68: 26-38, 2014.

Biographies

Razieh Asgarnezhad received her B.Sc. and MSc. degrees in Computer Engineering from Kashan Azad University in 2009 and Arak Azad University in 2012, respectively. She is currently a Ph.D. candidate at the Department of Computer Engineering at Isfahan Azad University. Her current researches include Data Mining, Text Mining, Learning Automata, and Wireless Sensor Network.

S. Amirhassan Monadjemi is an Associate Professor at the University of Isfahan and Senior Lecturer, School of continuing and lifelong education at the National University of Singapore. He received a Ph.D. in Pattern Recognition from the University of Bristol in 2004. His research interests include Artificial Intelligence, Machine Vision, and Data Analysis.

Mohammadreza SoltanAghaei is an Associate Professor at the Islamic Azad University of Isfahan. He received a Ph.D. in Computer Network from UPM University of Malaysia in 2010. His research interests include Computer Networks and Data Mining.

How to cite this paper:
R. Asgarnezhad, A. Monadjemi, M. SoltanAghaei, “NSE-PSO: Toward an Effective Model using Optimization Algorithm and Sampling Methods for Text Classification,” Journal of Electrical and Computer Engineering Innovations, 8(2): 183-192, 2020.

DOI: 10.22061/JECEI.2020.7295.379

URL: http://jecei.sru.ac.ir/article_1460.html