Decision tree combined with PSO-based feature selection for sentiment analysis

IOP Conf. Series: Journal of Physics: Conf. Series **1196** (2019) 012018 doi:10.1088/1742-6596/1196/1/012018

Decision tree combined with PSO-based feature selection for sentiment analysis

Rifkie Primartha[a], Bayu Adhi Tama[b⋆], Azhary Arliansyah[a], Kanda Januar Miraswan[a]

[a]Department of Informatics Engineering, Sriwijaya University, Palembang, Indonesia
[b]School of Management Engineering, Ulsan National Institute of Science and Technology, Ulsan, Rep. of Korea

E-mail: {rifkie, kandajm}@ilkom.unsri.ac.id; bayu@unist.ac.kr; arliansyah_azhary@yahoo.com
⋆corresponding author

Abstract. Sentiment analysis can be considered as a classification task in natural language processing as it harnesses classification algorithm to predict a particular class in a text data. In the classification task, feature extraction is a process to extract the features of the data so that it can be used as the input of the classification algorithm. However, not all features are particularly relevant for a classifier. Irrelevant features might significantly decrease the performance of classification algorithm. This paper proposes a PSO-based feature selection, combined with decision tree algorithm (PSO-C4.5) for sentiment analysis. The PSO-C4.5 is validated on a private data set, which is a sentiment data set about online transportation in Indonesia. The proposed method considerably enhances the performance of decision tree in comparison with the baseline.

1. Introduction

Sentiment analysis is the computational study about people's opinions, attitudes and emotions toward an entity. The entity might represent individuals, events or topics. These topics are most likely to be incorporated by reviews [1]. Sentiment analysis is considered as a classification task since it employs classification algorithm or classifier to discover the pattern in text data. In order to classify the class sentiment of the text data, a decision tree or other classification algorithms are frequently used. The decision tree is tree construction algorithm proposed by Ross Quinlann that can be used to build a decision tree as the classification model [2], [3].

The features are used as the inputs of the classifier to predict the sentiment class label. Term Frequency-Inverse Document Frequency or TF-IDF is utilized to extract features from text data set [4]. Text data sets generally yield a considerable high dimensional feature space, where the features are commonly represented in. Moreover, not all features are used as the inputs of classifier. The irrelevant features are deemed to result a worst accuracy for the classifier, while increasing the computational time [5]. Particle swarm optimization (PSO) is used to select only relevant features so the classification accuracy could be enhanced or at least to choose the features that would not decrease the classification accuracy.

The research conducted by Tsai et al [6] in 2012 shows that the PSO can optimize decision tree (C4.5) algorithm for backpropagation neural network (BPNN), and support vector machine

(SVM). The result shows the accuracy of decision tree is 70.99%, PSO-C4.5 is 75%, BPNN is 58.71%, LR is 74.08%, and SVM is 69.28%. A non-optimized C4.5 still outperforms BPNN and SVM, whilst PSO-C4.5 outperforms LR.

The work presented in [7] also shows that the PSO can be used as the one of feature selection method to optimize the performance of a classifier. The Reuters-21578 data set is used in the experiment, whilst the PSO are benchmarked with various feature selection methods, i.e.genetic algorithm, information gain rankings and CHI. They take into account k-Nearest Neighbor (k-NN) as the classifier and measure the macro-F1 and micro-F1 using precision and recall metrics. Currently, a combination of PSO-based feature selection and random forest has shown a promising result for anomaly-based intrusion detection system [8]. The proposed technique statistically outperforms deep neural network and rotation forest in terms of accuracy, precision, and recall performance metrics.

Motivated by the above-mentioned outcomes, in this paper, we employ PSO-based feature selection, as well as a decision tree algorithm for sentiment analysis. The proposed method is used to solve multi label classification problem in sentiment analysis, i.e. positive, negative, and neutral. The reminder of the paper is broken down in the following parts. Section 2 briefly describes material (data set) and methods used in our work. Section 3 details experimental results and discussion and finally this paper is summarized in Section 4.

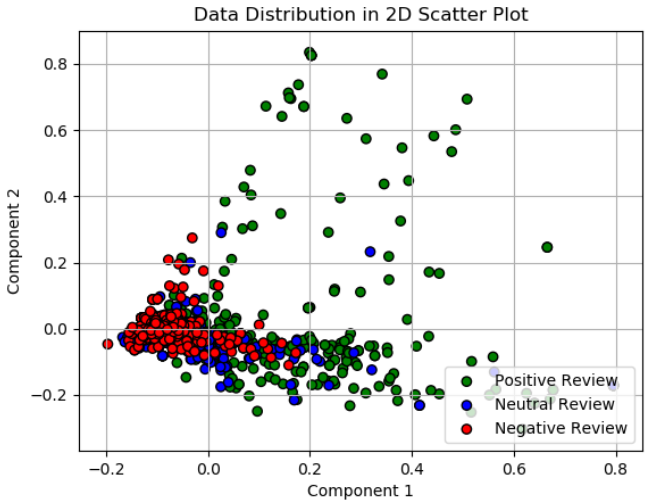

Figure 1. Data set visualization in 2D scatter plot

2. Material and Methods

2.1. Data set

The data set used in this paper is the user review of on-line transportation in Indonesia. We gather the data by conducting a survey. The data set contains reviews with three class sentiments, i.e. positive, negative, and neutral. Total samples of 1011 reviews are obtained, consisting of 344 positive reviews, 368 negative reviews, and 299 neutral reviews. We preprocess the data using text preprocessing techniques, i.e. *casefolding*, *tokenizing*, *stopword* removal, and *stemming*. We use *Sastrawi* package in Python to preprocess the Indonesian text.

We visualize the data to show data points distribution and outliers. The n-dimensional features are reduced into two and three dimensional space using Principal Component Analysis

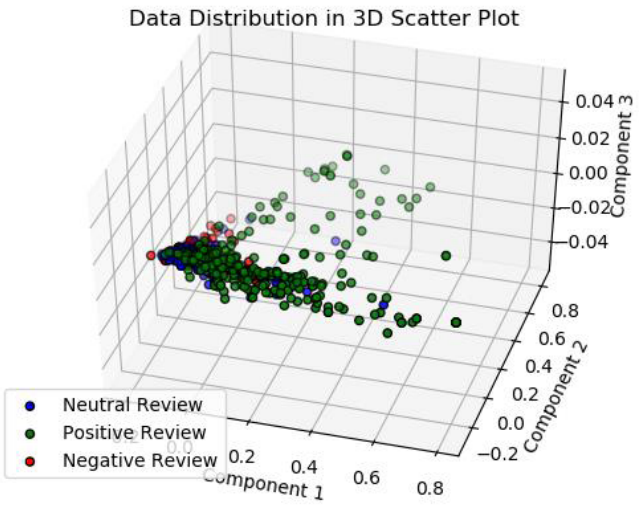

Figure 2. Data set visualization in 3D scatter plot

(PCA), so that it can be visualized into two and three dimensional plot, respectively [9]. The data points distribution can be visualized in two and three dimensional space, as depicted in Figure 1 and Figure 2, respectively.

2.2. Methods

In this section, we briefly discuss feature extraction and feature selection used in our experiment. Feature extraction is the process to extract features from data set. Many feature extraction methods are available based on what type of data set is used for classification task. For text data sets, we used TF-IDF to extract the features. TF-IDF is used as a weighting scheme for text by counting the frequency of each term in text document using Term Frequency (TF) and counting the relevance of a term within a collection of documents using Inverse Document Frequency (IDF).

The features obtained from TF-IDF, then would be selected by PSO, where the features are denoted as the particle position in the search space. PSO is a population-based evolutionary computation technique developed by Kennedy J and Eberhart R [10], particle position is represented with binary value 0 or 1. Zero means the feature is not chosen in the particle, whilst one means otherwise [11]. Therefore, if the feature space contains n-dimensions, the position of the particle is represented with n binary digits.

Next, we train the classifier with only selected features in the particle and then test it to get the accuracy of the classifier. The accuracy is considered as the solution or fitness value for the particle, whilst a decision tree algorithm is used as the objective function.

To run PSO, first of all, we have to determine the parameters value. The PSO parameters are the population size, iteration size, C_1, C_2, and the target. The position of each particle in the population is randomly generated with 0 or 1. Each particle has velocity represented in n-dimensions. Initially, the velocity of each particle in each dimension is 0. Each particle also has inertia weight initialized uniformly with a random number between 0 and 1.

3. Result and Discussion

There are several processes in this experiment to optimize decision tree algorithm using PSO, i.e. preprocessing, feature extraction using TF-IDF, feature selection using PSO, training the classifier, and validation test using 10-fold cross validation. We use decision tree algorithm and PSO implemented in Python. The preprocessing is used to remove some noises or outliers in data set. The process is undertaken by using *Sastrawi* package that includes $casefolding$, $tokenizing$, $stemming$, and $stopword$ removal.

Feature extraction is the process to extract the features from data set. This process is done by using $CountVectorizer$ and $TfidfTransformer$ modules in $scikit-learn$ package. Those two modules implement TF-IDF. Validation test is the process of measuring the performance or accuracy of the classifier, which is carried out by using a one-round k-fold cross validation. The cross validation splits the data set into k disjoint folds or partitions. On the k subsamples, one fold is used as testing samples, and the rest $(k-1)$ subsamples are used for training. This process is then repeated k times. We consider $k = 10$, which is so-called 10-fold cross validation. We discuss the experimental results by comparing the original decision tree algorithm and the proposed approach (PSO-C4.5) as following.

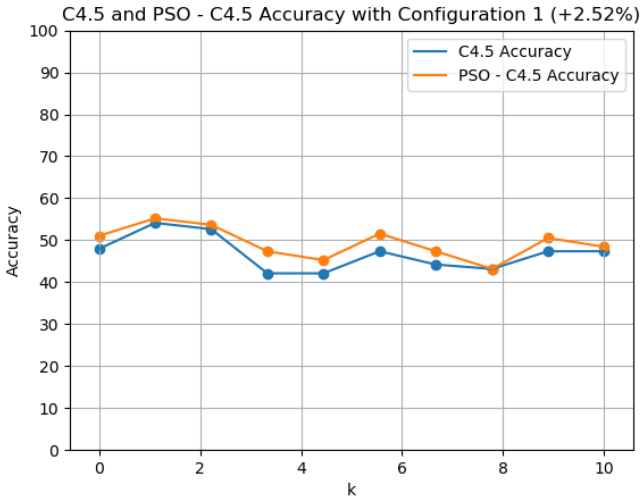

Figure 3. Accuracy of C4.5 and PSO C4.5 in the first experiment

We conduct three experiments by taking into consideration different parameters of PSO. In the first experiment, we set the parameters of PSO as follows. Population and iteration Size are set to 20, whilst C_1 and C_2 are set to 0.7 and 0.5, respectively. The performance accuracy for each fold in the first experiment is presented in Figure 3. In the second experiment, the PSO parameters are chosen as follows. Population size is set to 20 with 40 iteration size. The value of C_1 and C_2 are the same with the previous experiment. The result of the second experiment is shown in Figure 4. Finally, in the third experiment, we include the parameter of PSO as follows. Population size is increased to 40, whilst other parameters, i.e. iteration size, C_1, and C_2 are the same as the first experiment. We exhibit the result of the third experiment in Figure 5.

In experiment 1 with its configuration, we can see that the average improvement of C4.5 is 2.52% with the best result is 5.26% obtained from the 4th fold. The average improvement of C4.5 in experiment 2 is 1.89% with the best improvement is 5.26% occurred in the 9th fold. Lastly, it is evidenced that the average increase of accuracy of C4.5 in the 3rd experiment is 2.52% with the highest increase 5.26% occurred in 9th fold. From the result, it can be concluded

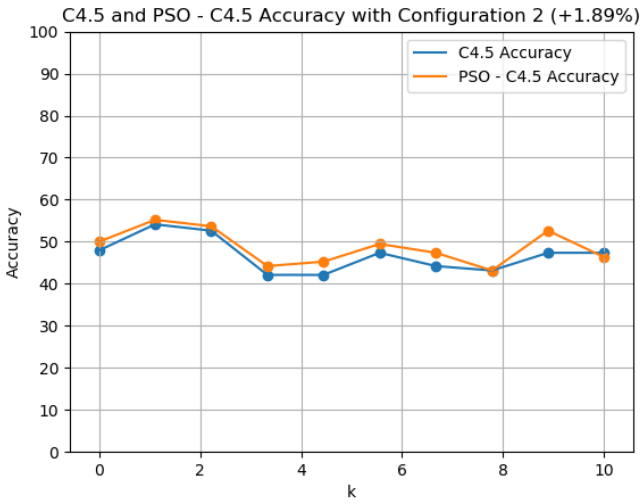

Figure 4. Accuracy of C4.5 and PSO C4.5 in the second experiment

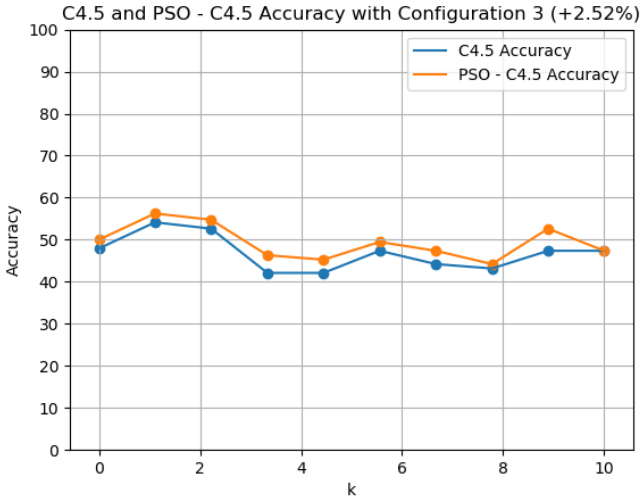

Figure 5. Accuracy of C4.5 and PSO C4.5 in the third experiment

that PSO can be used to optimize C4.5 for sentiment analysis by feature selection although the data set contains outliers with high dimensional features. This study suggests the importance of feature selection for sentiment classification task.

4. Conclusion

In this paper, we combined a decision tree classifier with PSO-based feature selection for sentiment analysis. A number of experiment scenarios were performed by considering different parameter setting of PSO. The proposed method significantly outperforms the performance of original decision tree algorithm in terms of accuracy metric. This paper revealed that PSO-based feature selection and decision tree classifier are the promising methods for sentiment analysis.

For future work, we will focus on the use of other state-of-the-art classification techniques, i.e. deep learning and ensemble methods to improve the prediction. Furthermore, we will use publicly benchmark data sets as a basis of our comparative study.

References

[1] Medhat W, Hassan A and Korashy H 2014 *Ain Shams Engineering Journal* **5** 1093–1113
[2] Quinlan J R 2014 *C4. 5: programs for machine learning* (Elsevier)
[3] Firdaus M A, Nadia R and Tama B A 2014 Detecting major disease in public hospital using ensemble techniques *International Symposium on Technology Management and Emerging Technologies (ISTMET)* (IEEE) pp 149–152
[4] Tang J, Alelyani S and Liu H 2014 *Data classification: Algorithms and applications* 37
[5] Leskovec J, Rajaraman A and Ullman J D 2014 *Mining of massive datasets* (Cambridge university press)
[6] Tsai M C, Chen K H, Su C T and Lin H C 2012 An application of pso algorithm and decision tree for medical problem *2nd International Conference on Intelligent Computational Systems (ICS'2012)* pp 124–126
[7] Aghdam M H and Heidari S 2015 *Journal of Artificial Intelligence and Soft Computing Research* **5** 231–238
[8] Tama B A and Rhee K H 2018 An integration of PSO-based feature selection and random forest for anomaly detection in IoT network *MATEC Web of Conferences* vol 159 (EDP Sciences)
[9] Liu S, Maljovec D, Wang B, Bremer P T and Pascucci V 2017 *IEEE Transactions on Visualization & Computer Graphics* 1249–1268
[10] Kennedy J 2011 Particle swarm optimization *Encyclopedia of machine learning* (Springer) pp 760–766
[11] Yang C S, Chuang L Y, Li J C and Yang C H 2008 Chaotic maps in binary particle swarm optimization for feature selection *Soft Computing in Industrial Applications, 2008. SMCia'08. IEEE Conference on* (IEEE) pp 107–112