EAI Endorsed Transactions
on Context-aware Systems and Applications Research Article EAI.EU

Opinion Mining with Density Forests

Phuc Quang Tran[1][0000−0001−5271−2323], Dung Ngoc Le Ha[2][0009-0004-6247-0355], Hanh Thi My Le[3][0000-0002-1386-1390], Hiep Xuan Huynh[4*,][0000-0002-9213-131X]

[1] Faculty of Foreign Language and Informatics People's Police College II, Ho Chi Minh city, Vietnam
[2] Can Tho University of Technology, Can Tho city, Vietnam
[3] University of Science and Technology Da Nang University, Danang, Vietnam
[4]Can Tho University, Can Tho city, Vietnam

Abstract

In this paper, we propose a new approach for opinion mining with density-based forests. We apply Density-Based Spatial Clustering of Applications with Noise (DBSCAN) to identify clusters of data points in a space of feature vectors that are important features of hotel and restaurant reviews, and then use the clusters to construct random forests to classify whether the opinions expressed about features in the reviews are positive or negative. Our experiment uses two standard datasets of hotel and restaurant reviews in two different scenarios. The experimental results show the effectiveness of our proposed model.

Keywords: DBSCAN Clustering, Density Forests, Opinion Mining, Hotel Reviews, Restaurant Reviews.

Received on 21 April 2023, accepted on 29 April 2023, published on 10 July 2023

doi: 10.4108/eetcasa.v9i1.3272

*Corresponding author. Email: hxhiep@ctu.edu.vn

1. Introduction

Opinion mining [2] is an area of computer science that involves natural language processing (NLP) and data mining related to views expressed in documents. The techniques used in opinion mining applications include machine learning algorithms, lexical-based approaches, etc. to classify opinions, determine whether the opinion is positive or negative. Unsupervised machine learning algorithms are also widely applied to the initial data processing to select important features and group them to improve the performance of the classifier. In paper[1] presented the new clustering algorithm DBSCAN relying to a density-based notion of clusters which is designed to discover clusters of arbitrary shape. The results of experiments demonstrate that DBSCAN is significantly more effective in discovering clusters of arbitrary shapes with high point density. In paper [11] presented an algorithm called best-scored clustering forest to efficiently solve the single-level density-based clustering problem. The algorithm selects a random tree with the best experimental performance from a number of completely random candidates. With the intrinsic advantage of random forest, this proposal achieves computational efficiency by taking full advantage of parallel computation. The paper [10] proposed borrowing some ideas from the DBSCAN algorithm, and a Random Forest classifier was used as the basic model to improve the efficiency of random forest classification on unbalanced data, thus improving the prediction performance, particularly for minority classes. The experimental results proved the ability of the proposed algorithm (DBRF) to solve the problem of classifying objects located on the class boundary.

At present, opinion mining techniques have not addressed the problem of ensemble/majority influence according to the importance of different sets/clusters of opinion based on the density forest approach. Especially for large datasets of opinion, the data points represent important features in the high-dimensional feature space and have complex shapes. The random forest[3] algorithm for opinion mining on food reviews [8][14] has been effective in classifying opinions extracted from them.

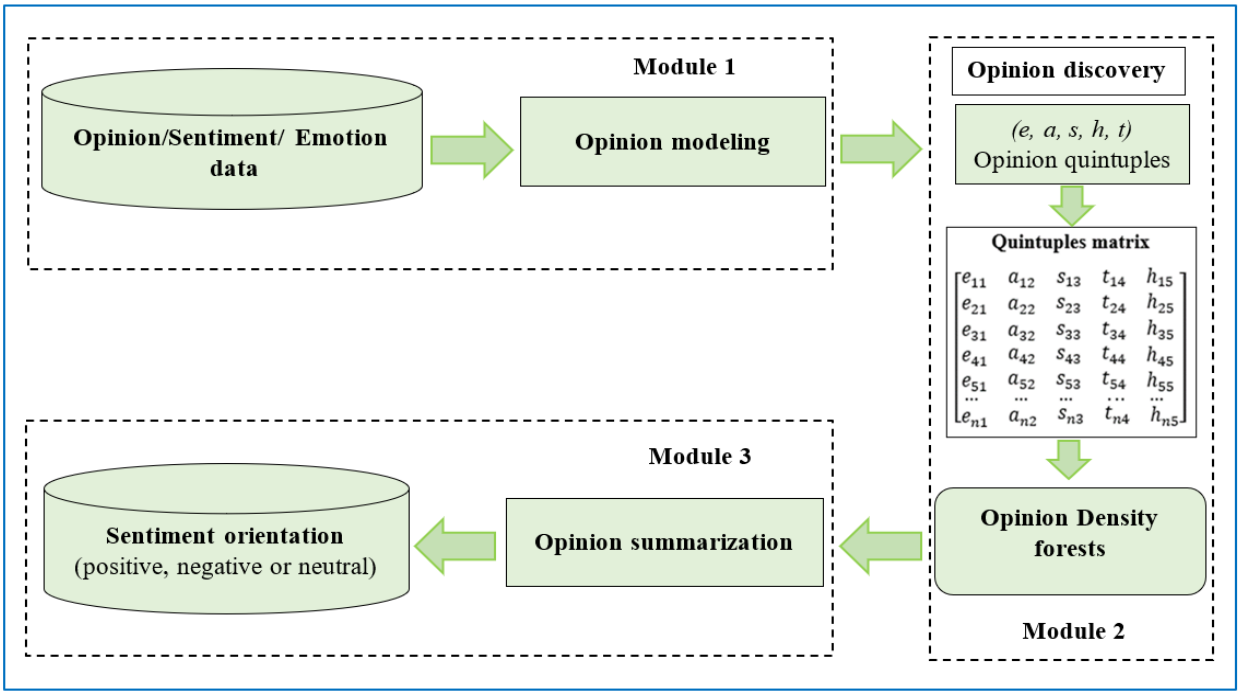

Figure 1. The proposed framework for opinion mining with density forests

In this paper, we propose a new approach for opinion mining based on the density forests in **Figure 1**. The problems affecting ensemble/majority opinion vary according to the importance of important sets/clusters. We constructed this model by approaching the DBSCAN clustering algorithms to cluster the important features of opinion data, including entity name, aspect name, reviewer name, and reviewer time, in a multidimensional feature vector space. The clusters obtained that build random forests are considered a base learner. The results of opinion polarization are aggregated according to the predicted majority vote of individual random forests.

This paper is organized as follows: Section 2 presents the problems related to opinion modeling; the proposed opinion discovery tasks are outlined in Section 3; Section 4 describes the opinion quintuple matrix; Section 5 proposes a model of the density-based forest for opinion mining; Section 6 describes the evaluation measures of the model; Section 7 provides an opinion summarization of the previous tasks, and Section 8 describes the experiments with two scenarios, discusses the results, and draws the conclusion of the paper.

2. Opinion Modeling

2.1. Opinion

An opinion [5] is modeled as a quadruple (g,s,h,t). where g is the sentiment target, s is the sentiment of the opinion about the target g, and h is the opinion holder which means the person or organization who holds the opinion, t is the time when is the time when the opinion is expressed.

Sentiment target

The sentiment target or opinion target [5] is an entity or a part or attribute of the entity that the sentiment has been expressed. An entity $\mathbf{e}$ is a product, service, topic, person, organization, issue, or event. It is described with a pair, e: (T,W), where T is a hierarchy of parts, subparts, and so on, and $\mathbf{W}$ is a set of attributes of $\mathbf{e}$. Each part or subpart also has its own set of attributes.

Sentiment of opinion

Sentiment of opinion [5] is the underlying feeling, attitude, evaluation, or emotion associated with an opinion. It is represented as a triple quadruple (y,o,i) where y is the type of the sentiment, o is the orientation of the sentiment, t is the intensity of the sentiment.

Opinion holder

The opinion holder [5] is the person or organization that expresses the opinion.

Time of opinion

The time of opinion [5] is the posting time when the opinion was expressed by the opinion holder.

2.2. Simplify Definition

An opinion [5] is a quintuple (e, a, s, h, t), where e is the target entity, a is the target aspect of an entity, e on which the opinion has been given, s is the sentiment of the opinion on aspect a of entity e, h is the opinion holder, and t is the opinion posting time; s can be positive, negative, or neutral, or a rating (e.g., 1–5 stars). In the case of opinion directed at a whole entity, the special aspect *GENERAL* is used to express that opinion. Here e and a together represent the opinion target.

2.3. Reason and Qualifier

A reason [5] for an opinion is the cause or explanation of the opinion. A qualifier of an opinion limits or modifies the meaning of the opinion.

2.4. Opinion document

An opinion document [5] d contains opinions about a finite set of entities $\{e_1, e_2, \ldots, e_r\}$ and a subset of aspects of each entity. The opinions are from a finite set of opinion holders $\{h_1, h, \ldots, h_p\}$ and are given at a particular time point t.

3. Opinion Discovery

Given an opinion document D, mining opinions consists of the following eight main tasks [5]:

Task 1. Entity Extraction and Resolution

Perform the task of extracting entities expressions in documents D and group entities synonyms into clusters (or categories). Each entity represents a clustering expression of the entity.

Task 2. Aspect Extraction and Resolution

Perform the task of extracting aspects expressions in document D and group aspects synonyms into clusters (or categories). Each aspect represents a clustering expression of the aspect.

Task3. Opinion Holder Extraction and Resolution

Extract the expression of the holder of each opinion from the review or structured data and group them.

Task 4. Time extraction and standardization

Extract the posting time of each opinion and standardize different time formats.

Task 5. Aspect Sentiment Classification or Regression

In the case of sentiment classification, determine the aspect (or entity) whose opinion is positive, negative, or neutral. In the case of regression, determine the numeric sentiment rating score of the aspect (or entity).

Task 6. Opinion Quintuple Generation

This task is generating all opinion quintuples (e, a, s, h, t), expressed in D from previous tasks.

Task 7. Opinion Reason Extraction and Resolution

Perform the task of extracting reason expressions for each opinion and group reason synonyms into clusters. Each reason represents a clustering expression of the reason for the opinion.

Task 8. Opinion Qualifier Extraction and Resolution

Perform the task of extracting qualifier expressions for each opinion and group qualifier synonyms into clusters. Each qualifier represents a clustering expression of the qualifier for the opinion.

4. Quintuples matrix

Given an opinion quintuple (e, a, s, h, t). The discrete values of the features in the set of opinion quintuple are converted to numerical values in the matrix, $n \times 5$, with n being the number of samples. A matrix of opinion quintuple is shown in **Figure 2**.

$$\begin{bmatrix} e_{11} & a_{12} & s_{13} & h_{14} & t_{15} \\ e_{21} & a_{22} & s_{23} & h_{24} & t_{25} \\ e_{31} & a_{32} & s_{33} & h_{34} & t_{35} \\ e_{41} & a_{42} & s_{43} & h_{44} & t_{45} \\ e_{51} & a_{52} & s_{53} & h_{54} & t_{55} \\ \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots \\ e_{n1} & a_{n2} & s_{n3} & h_{n4} & t_{n5} \end{bmatrix}$$

Figure 2. Quintuples matrix

The quintuples matrix is called a feature matrix as an input to the process of building an opinion density forest as the basis for mining and summarizing opinion.

5. Opinion Density Forests

A forest of opinion densities in **Figure 3** is a collection of randomly trained clustering trees, the leaves of which contain opinion prediction models.

5.1 Feature vectors

Feature vectors are converted from a quintuple matrix to be used for clustering feature points in a multidimensional vector space. The feature vectors were digitized from the values of the features to be used as input for clustering.

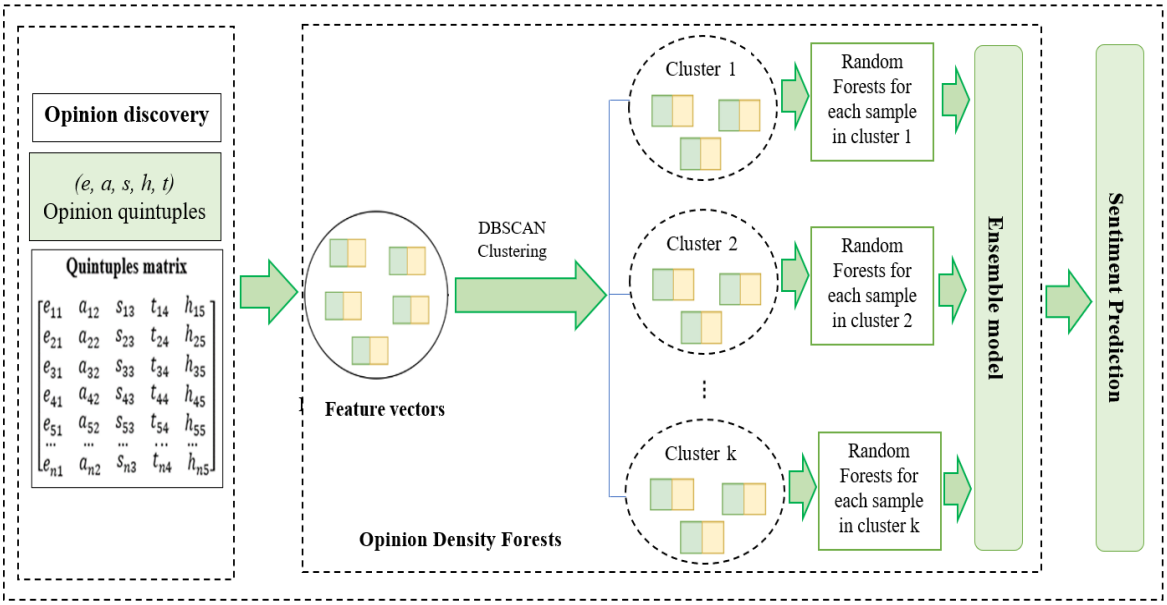

Figure 3. Illustrates proposed model of the density-based forest for sentiment prediction

5.2 DBSCAN Clustering

Let $O = (e, a, s, h, t) \in \mathbb{R}^d$, each point o is represented as a multidimensional feature vector. The output of the DBSCAN clustering is clustered k of varying densities.

There are two parameters that affect the clustering of the DBSCAN algorithm[1][6]: MinPts, amd eps (ε). MinPts and eps are non-empty subsets of O that satisfy the following conditions:

- $\forall o_1, o_2$: if $o_1 \in k$ and o_2 is density-reachable from o_1. Eps and MinPts, then $o_2 \in k$ (Maximality).
- $\forall o_1, o_2 \in k$: o_1 is density-connected to o_2. Eps and MinPts (Connectivity).

5.3 Opinion density forests model

For each cluster obtained, given a collection of points $O_0\{o\}$ is unlabeled used for training a density forest [9]. Each tree [12] is trained in the forest independently and employs randomized node optimization. Therefore, the optimization of the oth split node is done as follows:

$$\theta_j = \underset{\theta \in \tau_j}{\operatorname{argmax}} I(O_j, \theta) \tag{1}$$

With the general information increase is defined as

$$I(O_j, \theta) = H(O_j) - \sum_{i \in \{L,R\}} \frac{|O_j^i|}{|O_j|} H(O_j^i) \tag{2}$$

To determine the exact entropy $H(O)$ of a set of points O, as the data points have no labels, it is necessary to use an entropy that applies to unlabeled data. Assuming that the multivariate Gaussian distributions [13] are at the nodes of the clustering tree, then the differential (continuous) entropy difference can be expressed as follows:

$$H(O) = \frac{1}{2}\log((2\pi e)^d |\Lambda(O)|) \tag{3}$$

Where Λ is related to the $d \times d$ covariance matrix, the information gained from formula (2) is reduced to

$$I(O_j, \theta) = \log(|\Lambda(O_j)|) - \sum_{i \in \{L,R\}} \frac{|O_j^i|}{|O_j|} \log(|\Lambda(O_j^i)|) \tag{4}$$

With $|\Lambda(O_j)|$ indicating a determinant for matrix arguments, or cardinality for set arguments.

The tth tree in a forest has a set of leaves that define a data partition, with each leaf associated with a unique data point.

$$l(o): \mathbb{R}^d \rightarrow \mathfrak{L} \subset \mathbb{N} \tag{5}$$

Where $l(o)$ denotes the leaf reached deterministically by the input point o, and $\mathfrak{L}$ is the set of all leaves in a given tree. The tree index t is not shown here to avoid cluttering the notation. The statistics of all training points at each leaf node are summarized by a single multivariate Gaussian distribution $\mathcal{N}(o, \mu_{l(o)}, \Lambda_{l(o)})$. Then, the output of the tth tree is

$$p_t(o) = \frac{\pi_{l(o)}}{Z_t} \mathcal{N}(o; \mu_{l(o)}, \Lambda_{l(o)}) \tag{6}$$

The vector μ_l represents the mean of all the points reaching the leaf l, and Λ_l is the associated covariance matrix. The scalar π_l is the ratio of all training points that reach leaf l, i.e $\pi_l = \frac{|o_l|}{|o_0|}$

In formula (6), to ensure the normalization of probabilities, it is necessary to incorporate the Z_t partition function, which is defined as follows

$$Z_t = \int_o \left(\sum_l \pi_l \, \mathcal{N}(o; \mu_{l(o)}, \Lambda_{l(o)}) p(l|o) \right) do. \tag{7}$$

However, in a density forest, each data point reaches exactly one terminal node per tree. Thus, the condition $p(l|o)$is a delta function $p(l|o) = \delta(l = l(o))$, so formula (7) becomes

$$Z_t = \int_o \pi_{l(o)} \mathcal{N}(o; \mu_{l(o)}, \Lambda_{l(o)}) do. \tag{8}$$

The ensemble model of the forest density is calculated as the average of all tree densities, as follows

$$p(o) = \frac{1}{T} \sum_{t=1}^{T} p_t(o), \tag{9}$$

With a density of each tree $p_t(o)$

6. Evaluation

In the case of two-class classification of sentiment orientation as positive and negative, using the following confusion matrix[7]:

Table 1. Confusion matrix

	Predicted positive	Predicted negative
Actual positive	TP	FP
Actual negative	FN	TN

In **Table 1**, true positives (TP) indicate the number of samples in which the actual and predicted values are both positive. False positives (FP) indicate the number of samples in which the actual positive is predicted to be negative. True negative (TN) indicates the number of samples in which the actual result is negative, and the predicted result is also negative. False negative (FN) indicates the number of samples in which the actual result is negative, but the predicted result is positive. Evaluation of the sentiment classifier using the measures of accuracy, precision, recall, and F1-score.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{10}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{11}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{12}$$

$$F1 = \frac{2 * (Recall * Precision)}{Recall + Precision} \tag{13}$$

7. Opinion Summarization

Opinion summarization is a process of representing review information in a concise and summarized form. An opinion summarization [5] is referred to as an aspect-based opinion summary (or feature-based opinion summary). The aspect-based opinion summary about an entity e is in the following form:

1. *GENERAL*: number of opinion holders who are positive about entity e. Number of opinion holders who are negative about entity e.
2. *Aspect 1*: number of opinion holders who are positive about aspect 1 of entity e. Number of opinion holders who are negative about aspect 1 of entity e.
3. *Aspect n*: number of opinion holders who are positive about aspect n of entity e. umber of opinion holders who are negative about aspect n of entity e.

8. Experiment

8.1 Data Used

Our experiments on two datasets [15] about Yelp Filtered Reviews for opinion include text data on hotel and restaurance reviews. The database includes two files: yelpHotelData and yelpResData. The YelpHotelData dataset in the database includes three tables: Hotel,

Review, and Reviewer. The YelpResData dataset in the database includes three tables: Restaurant, Review, and Reviewer. The details are listed in the **Table 2, Table 3, Table 4, Table 5.**

Table 2. The total number of columns and rows in each table of the Yelp Hotel dataset

Tables	Total columns	Total rows
Hotel	13	283086
Reviewer	13	5123
Review	10	688329

Table 3. The statistics of the column names in each table the Yelp Hotel dataset

Hotel	Reviewer	Review
HotelID	ReviewID	Date
Name	Name	ReviewID
Location	Location	ReviewerID
ReviewCount	YelpJoinDate	ReviewContent
Rating	FriendCount	Rating
Categories	ReviewCount	UsefulCount
Address	FirstCount	CoolCount
AcceptsCreditCards	UsefulCount	FunnyCount
PriceRange	CoolCount	Flagged
WiFi	FunnyCount	HotelID
Website	ComplimentCount	
PhoneNumber	TipCount	
FilReviewCount	fanCount	

Table 4. The total number of columns and rows in each table of the Yelp Restaurant dataset

Tables	Total columns	Total rows
Restaurant	30	242652
Reviewer	13	16941
Review	10	788471

8.2 Preprocessing

From the initial dataset, the Yelp Hotel dataset includes three tables: Hotel, Reviews, and Reviewer, which have relationships with each other. We rely on the relationships between the data columns in the tables to combine these data tables into one. Then, important features in the selected data include the hotel name, aspects, reviewer, and reviewer time; this is achieved by converting the hotel aspects columns into rows and leaving the remaining columns unchanged. For the hotel aspects, we explore three aspects that affect the quality of the hotel: acceptance of credit cards (AcceptsCreditCards), room rate (PriceRange), and WiFi (WiFi). The results of the initial data conversion have 882,474 sets of opinion quintuples, the details of the features are listed in the **Table 6**.

Table 6. The statistics on the number of features selected in the Yelp Hotel dataset

Features of Hotel Reviews	Number of Different Values of Features
Entity (Hotel names)	123461
Aspect (Hotel aspects)	3
Holder (Hotel reviewer)	4596
Time (Reviewer time)	4382
Sentiment (Polarity)	5

Similarly, for the Yelp Restaurant dataset, the important features selected include the restaurant name, aspects, reviewer, and review time. In terms of restaurant aspects, we explored 19 aspects that influence the quality of the restaurant, such as being good for kids accepting credit cards, having parking, the attire of the staff, being good for groups, price range, taking reservations, delivery, takeout, waiter service, outdoor seating, WiFi, good for, alcohol, noise level, TV, ambiance, catering, and wheelchair accessibility. The results of the initial data conversion have 14,791,500 sets of opinion quintuples, the details of the features are listed in the **Table 7.**

Table 7. The statistics on the number of features selected in the Yelp Restaurant dataset

Features of Restaurant Reviews	Number of Different Values of Features
Entity (Restaurant name)	184167
Aspect (Restaurant aspect)	19
Holder (Restaurant reviewer)	12943
Time (Restaurant time)	4541
Sentiment (Polarity)	5

The sentimental value of the features is assessed by rating (from 1 to 5 stars), corresponding to 5 levels of polarization including “very negative”, “negative”, “neutral”, “positive”, and “very positive”. Our experiment is carried out on three levels of polarization: “negative”, “neutral”, and “positive”. To ensure that no data is lost, we combine samples with “very negative” and “negative” polarization into a “negative” sample, and pool samples with “very positive” and “positive” polarization into a “positive” sample.

8.3 Tool Used

We have programmed in R language for predictive model, the models used are: DBSCAN Clustering model, Random Forests model, and Stacking model. In addition, we also use libraries: library(ggplot2), library(stacks), library(tidymodels), library(h2o), library(randomForest), library(dbscan)[4], and to integrate models.

8.4 Scenario 1. The density forests for features include entity, aspect, holder, and time reviewers of hotel opinion data.

From the hotel review dataset, we perform DBSCAN clustering on the multidimensional feature vector space. The results of each clustering are used to build a random

Table 5. The statistics of the column names in each table the Yelp Restaurant dataset

Restaurant	Restaurant	Reviewer	Review
RestaurantID	Delivery	ReviewID	Date
Name	Takeout	Name	ReviewID
Location	WaiterService	Location	ReviewerID
ReviewCount	OutdoorSeating	YelpJoinDate	ReviewContent
Rating	WiFi	FriendCount	Rating
Categories	GoodFor	ReviewCount	UsefulCount
Address	Alcohol	FirstCount	CoolCount
Hours	NoiselLevel	UsefulCount	FunnyCount
GoodforKids	Ambience	CoolCount	Flagged
AcceptsCreditCards	HasTV	FunnyCount	RestaurantID
Parking	Caters	ComplimentCount	
Attire	WheelchairAccessible	TipCount	
GoodforGroups	WebSite	FanCount	
PriceRange	PhoneNumber		
TakesReservations	FilReviewCount		

classification forest to extract opinions on aspects of the hotel such as hotel name, hotel aspect, hotel reviewer, time of the reviewer. Random forests were built by us dividing the data in a clustering into *n* bootstrap samples. Each tree is built on a subset of random features which are the features of the hotel name, hotel aspect, hotel reviewer, reviewer time, and the decision feature which are the sentiment of opinion review features. We aggregate the classifier by taking the value that occurs most often for each classifier random forest. Random forests built on all clustering into a single classifier known as Density Forests to predict whether the polarity outcome of opinion is positive or negative.

Specifically, after preprocessing the hotel review dataset, we obtained 882,474 sets of opinion quintuples. We took a part that includes 112,275 opinion quintuples to the experiment. Next, split the data into 2 partitions as Train set (80%), and the Test set (20%). Then perform data clustering on the unlabeled Train Set. There are 7 clusters received from the DBSCAN algorithm in **Figure 4.** We took seven generated clusters to train random forests. The results by voting according to the majority to get the final result.

Table 8. The evaluation of the Density Forests model on the hotel reviews dataset

Accuracy	**Precision**	**Recall**	**F1**
0.92	0.91	0.60	0.72

The evaluation of the Density Forests on the hotel reviews dataset has an accuracy of 0.92 %, precision is 0.91 %, recall is 0.60%, and F1 is 0.72% in **Table 8.**

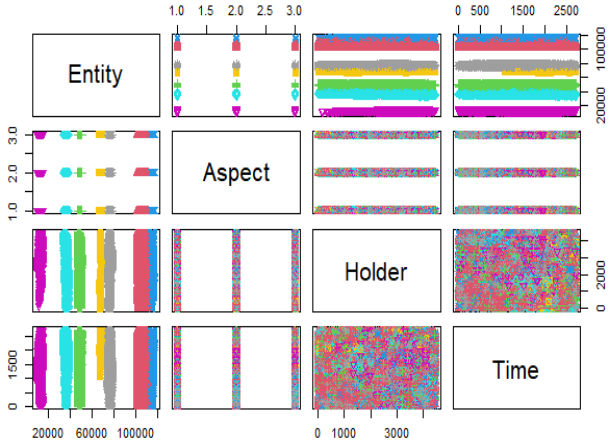

Figure 4. The results of DBSCAN clustering of hotel reviews dataset

8.5 Scenario 2. The density forests for features include entity, aspect, holder, and time reviewers of restaurant opinion data.

In the restaurant review dataset, the same process will be done in this part as compared to scenario 1. This dataset, after preprocessing, has 14,791,500 sets of opinion quintuples. We also present 112,275 opinion quintuples of the experiment for ease of following.

We first divide the initial data into 2 datasets including training dataset (80%), and test dataset (20%). The second is clustering on the unlabeled training dataset. The results of the clustering process are shown in **Figure 5**. There are 9 clusters received from the DBSCAN algorithm. The third is to train random forest models on 9 data clusters. Finally get the results by voting according to the majority of random forest models to get the final result.

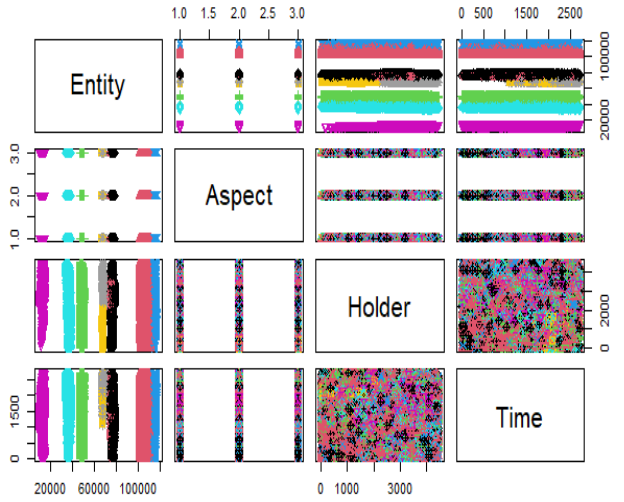

Figure 5. The results of DBSCAN clustering of restaurant reviews dataset

The evaluation results of the Density Forests training model on the restaurant review dataset in **Table 9** have an accuracy of 0.91%, precision is 0.89%, recall is 0.59%, and F1 is 0.71%.

Table 9. The evaluation of the Density Forests model on the restaurant reviews dataset

Accuracy	Precision	Recall	F1
0.91	0.89	0.59	0.71

9. Conclusion

We have proposed opinion mining based on the density forest approach. This model by approaching the DBSCAN clustering algorithms to cluster the important features of opinion data, including entity name, aspect name, reviewer name, and reviewer time, in a multidimensional feature vector space. The clusters obtained that build random forests are considered a base learners. The results of opinion polarization are aggregated according to the predicted majority vote of individual random forests each opinion polarity as positive or negative. Our experiment results on two standard datasets of hotel and restaurant reviews in two different scenarios show the effectiveness of our proposed model.

Acknowledgements.

The authors would like to thank distinguished professor Bing Liu, Department of Computer Science University of Illinois at Chicago (UIC) for sharing the dataset to conduct this study.

References

[1] Ester, M., Kriegel, H., Sander, J., Xu, X.. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96). ACM. 2016; 226–231.

[2] Liu, B., and Zhang, L. A survey of opinion mining and sentiment analysis. In Mining Text Data. Springer, Boston, MA. 2012; 415-463.

[3] Breiman, L.. Random forests. Mach. Learn. 2001; 45, 5–32.

[4] Hahsler, M., Piekenbrock, M., Doran, D.. dbscan: Fast Density-Based Clustering with R. Journal of Statistical Software. 2019; 91(1), 1–30.

[5] Liu, B..Sentiment Analysis: Mining Sentiments, Opinions, and Emotions. 2nd edn. Cambridge University Press, Cambridge. 2020.

[6] Weng, S., Gou, J., Fan, Z.. h-DBSCAN: A simple fast DBSCAN algorithm for big data. In Proceedings of Machine Learning Research 157, 2021.

[7] Zhou, Z.-H. Ensemble Learning: Foundations and Algorithms. Electronic Industry Press: Beijing, China, 2020.

[8] Phuc Quang Tran, Ngoan Thanh Trieu, Nguyen Vu Dao, Hai Thanh Nguyen and Hiep Xuan Huynh. Effective Opinion Words Extraction for Food Reviews Classification. International Journal of Advanced Computer Science and Applications(IJACSA). 2020; 11(7).

[9] Hongwei Wen, Hanyuan Hang: Random Forest Density Estimation. In Proceedings of the 39th International Conference on Machine Learning, PMLR 162:23701-23722, 2022.

[10] Dong, J. and Qian, Q.. A Density-Based Random Forest for Imbalanced Data Classification. Future Internet. 2022;14(90).

[11] Hang, Hanyuan, Cai, Yuchao and Yang, Hanfang: Density-based Clustering with Best-scored Random Forest. FOS: Computer and information sciences. 2019.

[12] Breiman, L., Friedman, J. H., Olshen, R. A., et al:. Classification and Regression Trees. CA: Wadsworth . 1984.

[13] Zhang, X: Gaussian Distribution. In: Sammut, C., Webb, G.I. (eds) Encyclopedia of Machine Learning. Springer, Boston, MA. 2011.

[14] Phuc Quang Tran, Hai Thanh Nguyen, Hanh My Thi Le, and Hiep Xuan Huynh. Ensemble Learning for Mining Opinions on Food Reviews. In Proceedings of the International Conference on Context-Aware Systems and Applications(ICCASA2021). 2021; pp 56–70.

[15] Arjun Mukherjee, Vivek Venkataraman, Bing Liu, and Natalie Glance. What Yelp Fake Review Filter Might Be Doing. In Proceedings of The International AAAI Conference on Weblogs and Social Media (ICWSM-2013), Boston, USA. 2013.