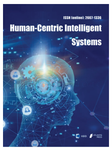
Human-Centric Intelligent Systems
Vol. **1(1–2)**; June (2021), *pp.* 32–42
DOI: https://doi.org/10.2991/hcis.k.210704.001; eISSN 2667-1336
https://www.atlantis-press.com/journals/hcis



Research Article

Semantic Knowledge Discovery for User Profiling for Location-Based Recommender Systems

Xiaohui Tao[1,*], Nischal Sharma[1], Patrick Delaney[1], Aimin Hu[2]

[1] *School of Sciences, University of Southern Queensland, Australia*
[2] *Guilin Tourism University, Guilin, China*

ARTICLE INFO

Article History

Received 12 March 2021
Accepted 04 July 2021

Keywords

Location-based recommender system
Foursquare
sentiment analysis
topic modelling
venue recommendation
places of interest

ABSTRACT

This paper introduces a purposed Location-based Recommender System (LBRS) that combines sentiment analysis and topic modelling techniques to improve user profiling for enhancing recommendations of Points of Interest (POIs). Using additional feature extraction, we built user profiles from a Foursquare dataset to evaluate our model and provide recommendations based on user opinions toward venues. Our combined model performed favourably against the baseline models, with an overall improved accuracy of 0.67. The limitations were the use of one dataset and that user profiles were constructed using predicted emotions extracted as features from review data with topic modelling, rather than literal user emotions. Nevertheless, this provides a step forward in user profile and emotion scoring, contributing further to the development of LBRS in the Tourism domain.

1. INTRODUCTION

Web-based platforms have transformed the tourism and hospitality industries, as travellers no longer wholly rely on travel professionals when finding a location [34]. A survey by Pang [24] indicates that 73%-87% of users who engage with these web-based tourism platforms (e.g. hotel or restaurant) are influenced by online reviews when selecting an intended destination. The content of these platforms consist of millions of ratings, threads, social media posts, and check-in information, which can create information overload. A Recommender System (RS) collects information on user preferences for a particular item and analyses them to predict user needs, providing personalized recommendations that may reduce overload and enable more efficient decisions [4].

The implementation of Recommender Systems in entertainment and e-commerce has shown progress, but there are still attempts to successfully apply this approach to the tourism and hospitality domain, where numerical ratings are difficult to collect and preferences toward domain items are too complex to be expressed as a scalar rating [16]. Location-based Social Networks (LBSNs) such as Foursquare, Gowalla, Facebook and Yelp encourage users to check-in, rate and comment on the locations they visited [5], but users of these platforms may not provide implicit ratings. A Location-Based Recommender System (LBRS) uses the data in LBSNs to predict user preference toward venues, then recommend a Point of Interest (POI). Many LBRSs consider only check-in data when modeling user preferences for a venue, which may be insufficient [6,38]. Although better than classical recommendations via user ratings, this fails to provide exact user preferences, as the data can create false assumptions about majorly checked-in venues (e.g. bad service at the venue), which cannot be extracted from check-in data alone [6].

For this study, we propose the development of a purposed LBRS that considers additional features from a dataset. Previous work has focused on non-textual attributes, such as categorical, temporal, spatial, and social activities. For example, Yang [36] considered sentiment analysis techniques, but did not consider semantic factors such as popular events, the most intriguing activity, environment or food. The similarity-based approach by Zheng [40] proposed user-based neighbour style explanations, which is similar to social network-based collaborative filtering approaches that give higher weight to friends of users, but does not consider user perspectives from their opinion or emotions [14,20]. Sentiment analysis for LBRS have been used in the past [36,42], but these were unable to capture the meaning of longer phrases properly and did not accurately capture the underlying phenomena presented in the data.

Our aim is to design an LBRS that employs both sentiment analysis and topic modelling to extract features from user reviews to better predict POIs. The purposed LBRS described in this paper analyses user reviews and performs sentiment analysis and topic modelling, on top of classical recommendation techniques. The ability of sentiment analysis to factor in emotions is considered more trustworthy than classical ratings by offering stronger judgement toward a venue, while topic modelling is becoming widely used as a probabilistic technique for text. The sentiment score of a user review combined with topic modelling provides stronger evaluation of a venue.

[*] *Corresponding author. Email: xiaohui.tao@usq.edu.au*

Peer review under responsibility of KEO (Henan) Education Technology Co. Ltd

The purposed LBRS for this research leverages text mining to quantify the polarity and semantics towards the main subject of a text. Our approach includes feature extraction of previously unconsidered topics such as certain events or attractions of the POI, which will help predict user preference by capturing the subjectivity in semantic orientation associated with a text [31]. In addition, we use feature extraction to create user profiles. Topic modelling is used to learn user topics from tips information associated with review data from Foursquare. Our objective is to reveal an unobserved context between people and locations to recover semantic information.

2. BASIS OF RECOMMENDER SYSTEMS

Recommender systems filter massive information by extracting data generated from a user's choices, interests and observed behaviour, which can be used to build his or her profile [13,26]. In research, a recommender system is broadly classified into three types: content-based filtering, collaborative filtering and hybrid systems.

Content Based Filtering (CBF) assumes that users who liked certain items in the past will also like new items with the same attributes. User profiles are automatically updated based on the feedback to individual items, and items are recommended to the user according to their profile [3]. It primarily relies on dictionary-bound relations between the terms used in user profiles and item content, so implicit associations between users are not considered [38]. *Collaborative Filtering* (CF) analyses user interactions and behaviours to determine their likes based on similar user choices. User-generated feedback can be used to detect taste commonalities between groups of users, then recommendations are generated based on inter-user similarities. CF locates the subset of similar users who have similar profiles and preferences, then provides suggestions on the assumption that users with similar tastes in the past will still have the same taste in future [3,21]. Memory-based algorithms use statistical techniques to determine a group of users who have a history of agreeing with a selected user, while model-based collaborative algorithms provide recommendations by firstly developing a model of user ratings. Finally, recommender systems that use a *Hybrid Filtering Approach* resolve the issues present in CF and CBF systems, such as cold start and gray sheep in CF and data sparsity and overspecialization problem in CBF [3]. Hybrid recommender systems combine different techniques under CF and CBF in a way that exploits the advantages and reduces the drawbacks. These can be implemented by separately making predictions and then integrating them, or by adding functions of the CBF system to a CF system, or by combining the two different models into one [27].

2.1. Social Networks Based Recommendations

Social network-based CF [14,20] gives higher weight of recommendations to friends of a user rather than a set of anonymous, similar users. As specified by Kautz [14], people who use SNs function as both "users" and "items". Thus, suggesting recommendations of potential friends or partners based on the bilateral nature of interaction in social networks. Research by Ye et al. [38] implemented a friend-based collaborative filtering (FCF) approach for location recommendations based on heuristics derived from observed geospatial characteristics in the Foursquare dataset, which was very effective. Similarly, Bao [5] found that social activity inferred from user data at a certain location also influences recommendations. Huang [18] proposed a method to automatically detect activities using the spatial-temporal attractiveness (STPA) of POI. The system exploits the connections between the user activities and profiles in a joint learning process [43]. Location Sequential Recommendations [44] use association rule-mining and sequence-mining over sequences of locations extracted from geo-tagged photos, GPS-trajectory. Based on the user's historical visiting pattern, the system creates an itinerary of scenic locations to visit that are popular among other users by preparing frequently travelled routes and implementing a route recommendation system [9].

2.2. Location Aware Recommendation

Location check-ins implicitly create a new social structure made up of inter-dependency between individuals derived from their locations. With inter-dependency, two persons co-occur in the same physical location, or share similar location histories and also determine the knowledge such as common interests, behaviour, and activities, inferred from user's history of locations and location-tagged data e.g. text, photos, etc. [7]. The generated data provides an idea of the spatial, social and temporal characteristics of how people use these platforms to model patterns of human mobility, which are significant factors for designing future location-based services [8].

Traditional RS for venues use a rating matrix. The matrix has one row for each user and one column for each item, with each element of the matrix showing the preference of a user toward an item. It is rare for all users to have rated all items, so the task of an RS is to predict the "missing" ratings, i.e. a users preference toward their un-rated items [33]. In user-based CF, a rating is estimated by aggregating the ratings of other users for item, while the aggregation is based on the user-user similarity. In contrast, item-based CF estimates ratings based on users past ratings on other items as well as item-item similarity [19,29]. Among the several User Item Rating Matrix models for LBRS, the most popular has to be the 0/1 scheme model by [38,39]. It takes 0 for non-visited locations and 1 for visited locations. By using the 0/1 model, authors such as Ye [38,39] consider the frequencies of check-ins and the social and geographic impact to compute user preference in POI.

Generic recommendations provide the user with the most popular venue, ignoring individual preferences, whereas a personalized approach provides users with the most suitable location by considering personalized recommendations with techniques like CF [36]. Zheng [40] proposed a personalized system to identify experienced travellers by applying a Hypertext Induced Topic Search (HITS) inference model over a Tree-Based Hierarchical Graph of users' historical trajectories. In this system, users with the same location history may have similarities and can be considered a friend. The connections inside the information network reflect users' geographical histories as well as their social relationships. Based on a sequence-matching algorithm that considers location hierarchies, the system finds users with similar travelling patterns [17]. Xiao [35] extends the user similarity approach by considering available semantic data, however, it primarily focuses on the user similarities and does not account for activity analysis or semantic score to predict the data, which our work encompasses.

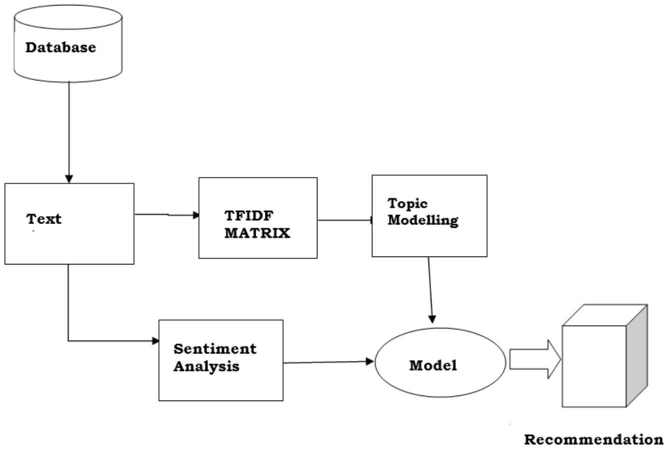


Figure 1 Architecture of purposed LBRS.

3. METHODOLOGY

For our purposed LBRS model, we capture the sentiment of a user "u" towards a location "l". A sentiment-enhanced user preference model, along with feature extraction, is implemented to create recommendations. Sentiments present in the texts assist us in identifying human opinion on certain topics determined from the corpus. The purposed work eliminates data sparsity by considering feature extraction and implementing a collaborative matrix factorization method. Thus, this research suggests a hybrid preference model by application of sentiment and semantic analysis in a location recommendation approach, which will create an effective and efficient recommendation. An overview of the Purposed LBRS is shown in Figure 1. The goal is to automatically recommend a POI by using a user sentiment score and topic modelling.

A user profile is a generic model that is built from data such as personal information, hobbies and travel history within the dataset, and arranged to show interactivity between a user and a system. Our purposed LBRS builds the user profile according to user venue preferences, including past travel, tips and reviews.

The model implements a user profile by creating a user model, shown in the stage before the recommendation of the POI in Figure 1. The research data did not have explicit user data, so implicit data generated is generated from user reviews to profile the user. The process of the LBRS implements sentiment analysis and topic modelling to construct a user profile from sentiments and topics classifications, in which a user clustering structure is created, forming a categorical dataset.

3.1. Data Pre-Processing

The dataset for our experiment is from the Foursquare API. Foursquare provides personalized recommendations of places to go near a user's current location, based on their previous browsing and check-in history [15]. This dataset contains: (a) user's tips and check-in data and (b) venue categorization. Location Data and User Opinions are available publicly by crawling the Foursquare-tagged tweets of the Twitter Public Feed, which is provided by Yang et al. [36]. The information collected does not interfere with user privacy and only acknowledges data made public by the users. Recent reviews from New York and London are extracted. The reviews of POIs from Foursquare are sourced from [36,37].

Table 1 Details of Foursquare database

	Venue	User
Count	3298	3112
Unique	2706	924
Top	3184	82569
Frequency	23	239

Data is considered from users who have performed one check-in per week and did not show sudden movement from a locale. The Foursquare dataset includes check-in, tip and tag data of restaurant venues in NYC collected between the 24th of October 2011 to the 20th of February 2012. It contains 3112 users and 3298 venues, with 27149 check-ins and 10377 tips. While it is diverse, it has a low sparsity value. We only considered data from users with 3 or more reviews in the database. With these parameters, a total of 924 unique users are identified, with a total of 2706 unique venues. Table 1 provides a snapshot of the Count, Unique values, Top review counts and Frequency of reviews per Venue and User.

Creation of a user model is required for the hybrid approach. The dataset did not have explicit user information, so we rely on implicit data generated by user reviews to develop the user profiles, including personal data (name, age, gender), hobbies (likes, dislikes), travel history, location data and certain topics. As stated previously, only data from users who provided at least 3 reviews toward a venue are considered, to mitigate the data sparsity problem. The largest number of reviews given by a single user was 23.

Data was cleaned using tokenization, with stemming and lemmatization [12]. Noise such as hashtags, punctuation, special characters, reluctant words, along with other irrelevant data, were removed. Pre-processing and cleaning also involved the removal of Unicode characters outside of ASCII, deleting duplicate reviews and non-alphabet characters, and conversion of all words to lowercase. Finally, stop word removal and stemming was conducted, in accordance with [22]. After pre-processing, our dataset consisted of 7402 reviews.

3.2. Sentiment Analysis of the Reviews

We extracted meaningful text from the corpus to produce clean reviews for the training and test sets. This information was also used for building the recommendation system [22]. We used the unsupervised algorithm SentiStrength for its dictionary-based approach of determining sentiment strength of a document [1]. Input text was split into sentences, with each tested to see if it contains a subjective or objective sentiment, then we extracted the object about which the opinion is expressed.

For each text, the SentiStrength output is two integers: 1 to 5 for positive sentiment strength, and a separate score of 1 to 5 for negative sentiment strength. Here, 1 signifies no sentiment and 5 signifies a strong sentiment score of the respected types. For example, a text with a score of 3 for positive and 5 for negative would contain moderate positive sentiment and strong negative sentiment. A neutral text would be coded as 1, 1. Two scales are used because even short texts can contain both positivity and negativity. The goal is to detect the sentiment expressed rather than its overall polarity [32].

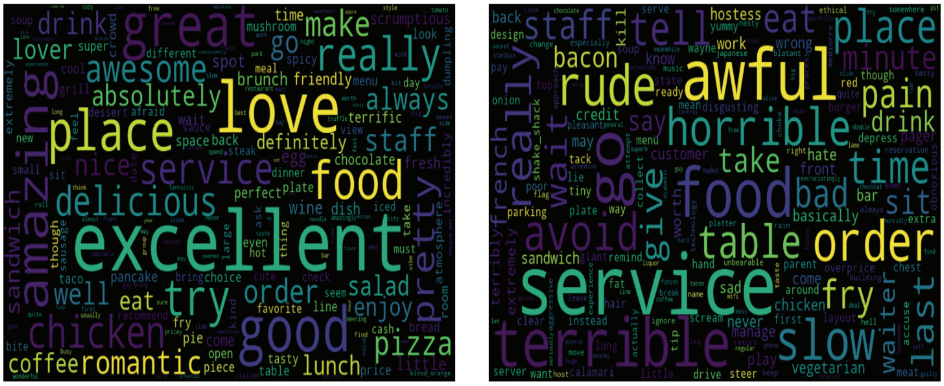


Figure 2 Positive words vs negative words.

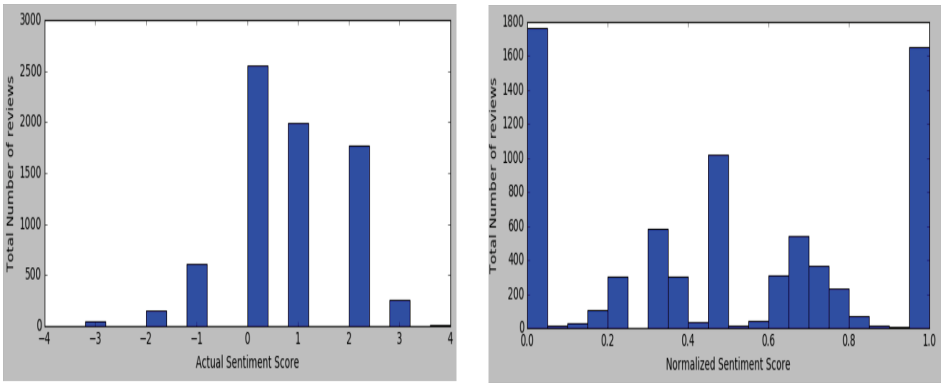


Figure 3 Actual vs normalized sentiment scores.

Positive suggestions were found to be 1.5 times higher than negative suggestions in the dataset. Figure 2 offers a comparison between the most Positive words and the most Negative words in the corpus.

From the overall range of 1 to 5 for both negative and positive sentiment score, we calculated a scale from -4 to 4 by summing the value of the sentiments. The sentences with -4 score are the Strongly Negative sentences (they had the most amount of negative words), 0 is assigned Neutral (reviews address both the negative and positive side equally), and the sentences with a score of 4 are Strongly Positive (no complaints in the sentence).

Within the reviews, some users were too lenient and some were too aggressive in providing reviews. So we scaled the user ratings from 0 to 1, with 0 being most negative and 1 being most positive. While normalizing user reviews scores, we found that positive and negative score are evenly distributed among the total users reviews. We converted the scales to binary measures as 'recommendation' for a positive opinion and 'no recommendation' for a negative opinion. A comparison between the actual sentiment scores and the normalized sentiment scores are shown in Figure 3.

3.3. Topic Modelling

With our technique, we extracted information contained in review text and converted these into appropriate topics. Rather than representing a text 'T' in its feature space, we represented 'T' in its topic space. Topic Modelling is used for recommending a location with a topic structure similar to the location the user has already visited.

We used Term Document Frequency (TD-IDF) to analyse our normalized data and adopted Latent Dirichlet Allocation (LDA) as our topic modelling method to review texts. By using TF-IDF over bag of words, we significantly reduced the sparsity of the feature matrix from 0.57% to 0.45%, which is essential for a good LDA model. This technique was used for semantic feature extraction, where IDF represents $\log(N/n)$, with N as the number of documents and n as the number of documents a term t has appeared within.

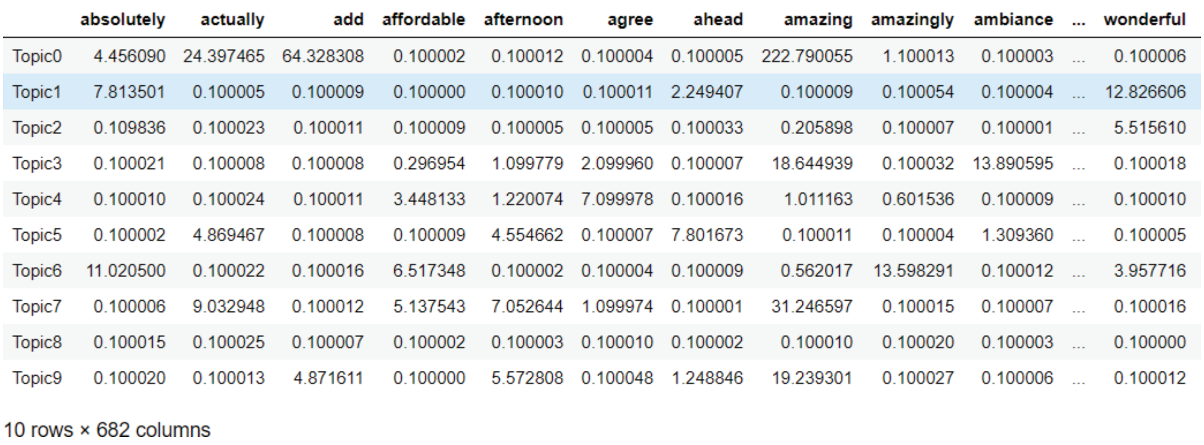
It can be mathematically represented as:

$$w_{i,j} = tf_{i,j} \cdot \log \frac{N}{df_i} \tag{3.1}$$

where, $t_{i,j}$ is number of occurrence in i and j, df_i is number of documents containing i and N is number of documents j.

After TF-IDF term frequencies were calculated, a Document-term matrix is created, showing TF-IDF weight for each word in a given review. This matrix is composed of every document as rows and every term as columns. This matrix is shown in Figure 4, with some examples of weighted words.

Considering each review as a document, we create one for every user. LDA is used to train our model for the constructed corpus of each POI. Our LDA model is built using the python library, scikit-learn [25]. With initial modelling, we selected 20 topics for our experiments, with 10 learning iterations and a batch size equal to 128 using all the CPU power. Following this, we calculated perplexity and log-likelihood to determine how accurately the produced model represents the statistics of the provided data.



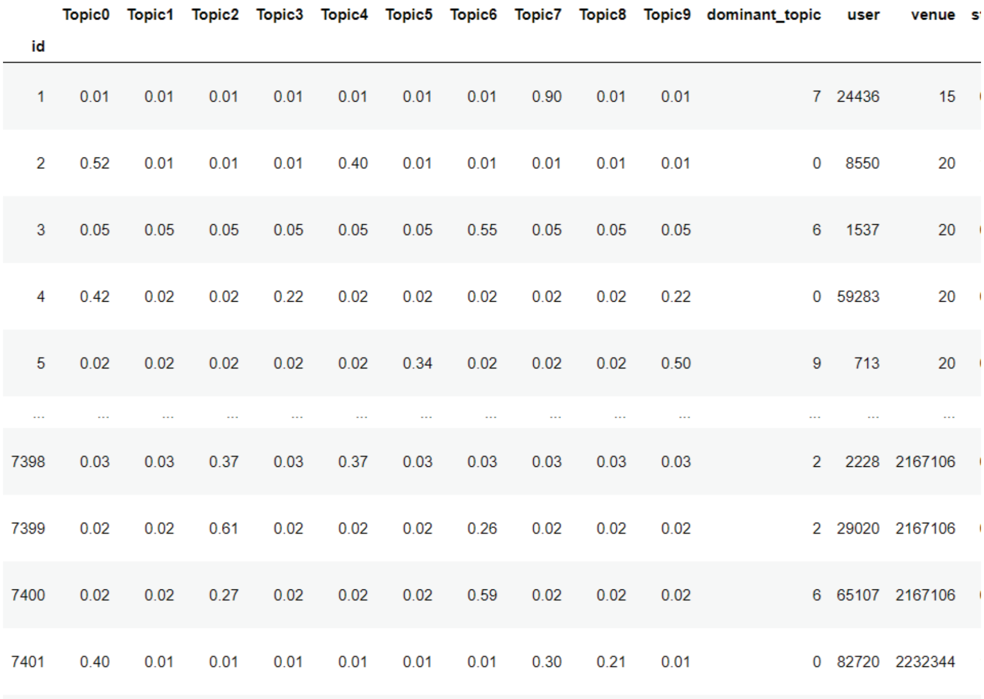
	absolutely	actually	add	affordable	afternoon	agree	ahead	amazing	amazingly	ambiance	...	wonderful
Topic0	4.456090	24.397465	64.328308	0.100002	0.100012	0.100004	0.100005	222.790055	1.100013	0.100003	...	0.100006
Topic1	7.813501	0.100005	0.100009	0.100000	0.100010	0.100011	2.249407	0.100009	0.100054	0.100004	...	12.826606
Topic2	0.109836	0.100023	0.100011	0.100009	0.100005	0.100005	0.100033	0.205898	0.100007	0.100001	...	5.515610
Topic3	0.100021	0.100008	0.100008	0.296954	1.099779	2.099960	0.100007	18.644939	0.100032	13.890595	...	0.100018
Topic4	0.100010	0.100024	0.100011	3.448133	1.220074	7.099978	0.100016	1.011163	0.601536	0.100009	...	0.100010
Topic5	0.100002	4.869467	0.100008	0.100009	4.554662	0.100007	7.801673	0.100011	0.100004	1.309360	...	0.100005
Topic6	11.020500	0.100022	0.100016	6.517348	0.100002	0.100004	0.100009	0.562017	13.598291	0.100012	...	3.957716
Topic7	0.100006	9.032948	0.100012	5.137543	7.052644	1.099974	0.100001	31.246597	0.100015	0.100007	...	0.100016
Topic8	0.100015	0.100025	0.100007	0.100002	0.100003	0.100010	0.100002	0.100010	0.100020	0.100003	...	0.100000
Topic9	0.100020	0.100013	4.871611	0.100000	5.572808	0.100048	1.248846	19.239301	0.100027	0.100006	...	0.100012

10 rows × 682 columns

Figure 4 Weightage of word for each topic.

Table 2 Topics

Category	Top 15 Words
Topic 1	*order, chicken, fry, wait, line, pork, long, come, bread, table*
Topic 2	*food, lunch, special, great, bad, staff, service, good, drink, nice*
Topic 3	*delicious, make, sandwich, coffee, small, cold, wrong, friend, little, menu*
Topic 4	*restaurant, amazing, know, eat, food, slice, good, list, time, choice*
Topic 5	*place, good, love, sit, bar, night, food, great, fresh, check*
Topic 6	*burger, chocolate, dumpling, die, worth, better, cheap, pancake, plate, shake*
Topic 7	*spicy, salad, tea, noodle, make, cocktail, sweet, perfect, mushroom, sure*
Topic 8	*great, pizza, spot, service, really, food, coffee, beer, favorite, brunch*
Topic 9	*good, best, ask, pretty, coffee, city, roll, french, free, order*
Topic 10	*try, sauce, hot, taste, egg, awesome, amazing, fry, grill, chicken*



	Topic0	Topic1	Topic2	Topic3	Topic4	Topic5	Topic6	Topic7	Topic8	Topic9	dominant_topic	user	venue	s
id														
1	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.90	0.01	0.01	7	24436	15	
2	0.52	0.01	0.01	0.01	0.40	0.01	0.01	0.01	0.01	0.01	0	8550	20	
3	0.05	0.05	0.05	0.05	0.05	0.05	0.55	0.05	0.05	0.05	6	1537	20	
4	0.42	0.02	0.02	0.22	0.02	0.02	0.02	0.02	0.02	0.22	0	59283	20	
5	0.02	0.02	0.02	0.02	0.02	0.34	0.02	0.02	0.02	0.50	9	713	20	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
7398	0.03	0.03	0.37	0.03	0.37	0.03	0.03	0.03	0.03	0.03	2	2228	2167106	
7399	0.02	0.02	0.61	0.02	0.02	0.02	0.26	0.02	0.02	0.02	2	29020	2167106	
7400	0.02	0.02	0.27	0.02	0.02	0.02	0.59	0.02	0.02	0.02	6	65107	2167106	
7401	0.40	0.01	0.01	0.01	0.01	0.01	0.01	0.30	0.21	0.01	0	82720	2232344	

Figure 5 Weightage of topic for each document.

Each topic contains specific words from which we identify the user and venue separately. This provides both a user profile based on the venue being mentioned in the reviews, and the preference toward a venue if they appear in the same topic. This follows an approach similar to Aciar et al. [2], in which consumer views are represented through clustering venues together.

To tune our LDA model, we selected the optimal number of topics to improve model efficiency. We did not regard the θ and ψ as parameters calculated directly, but considered the word posterior probability P(w—z) for the themes using Grid Search [25] to obtain the value of parameters (θ and ψ) indirectly. We observed that a significantly low optimal number of distinct topics (10), might

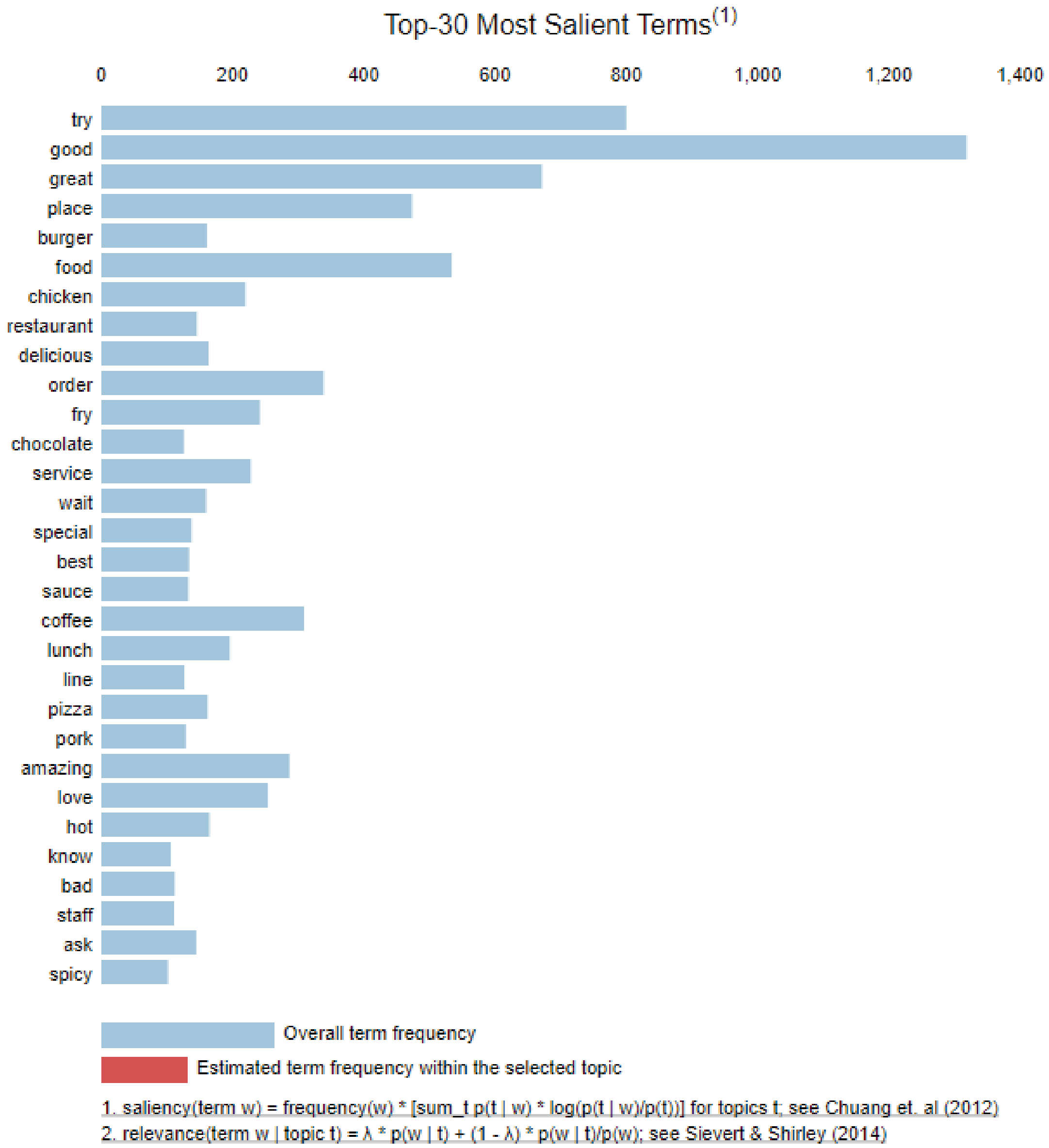


Figure 6 | Most common words.

be effective for classifying this corpus. Plotting the log-likelihood scores against number of topics shows the number of topics when using 10 has better scores and the Perplexity is also significantly low. Table 2 shows the top 10 topics extracted from reviews linked with restaurants and the top 15 words with the highest weight for each topic.

These topics share different weight in each document and the topic with highest weightage is considered the dominant topic of the particular document. In Figure 5, we can observe the weight of each topic across the 7401 unique documents. The dominant_topic column indicates which Topic was the heighest weight for its particular document, to give us a sense of the most frequently occurring topics as recommendations.

The histogram in Figure 6 presents the top 30 most salient terms from our dataset. Among all conversations in the data, Topic '0' has been interacted with more than 1200 times, while the least interacted topic is "5", which was interacted with less than 500 times. We used these topic weights for each document for our *k*-means experiment to properly classify the documents.

3.4. Document Classification Using *k*-Means

To categorize a document to a topic, we implement *k*-means clustering based on the document-topic probability matrix [11]. Since our best LDA model has 10 clusters, we set this as the number for our *k*-means experiment. We require the X and Y axis to draw the scatter plot, which is obtained through Singular Value Decomposition (SVD) [41]. SVD on the document topic matrix object requires total components equal to 2. The combination algorithm for topic detection in the Foursquare dataset is described as follows [23]:

1. Minimize the extent of the word-document matrix using SVD to form a diminished form of latent semantic-review matrix.
2. Group reviews in the minimised form using *k*-means clustering.

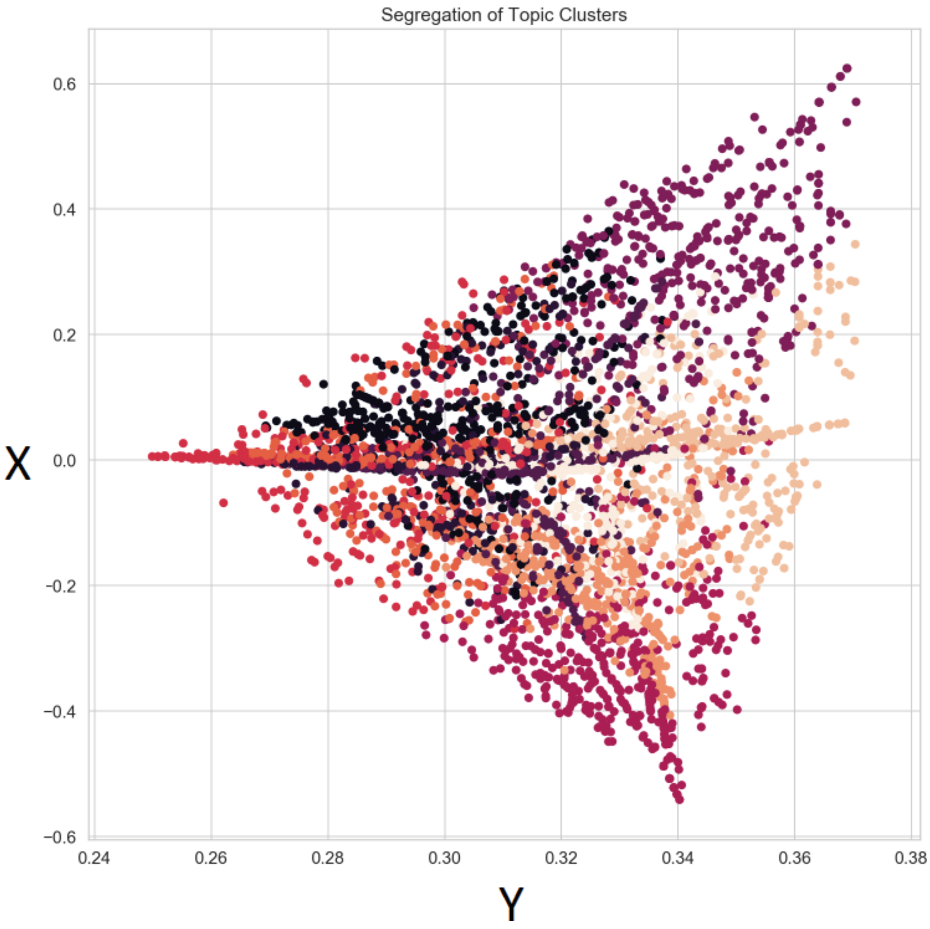


Figure 7 *k*-means clustering of data based on SVD.

Table 3 Details of Foursquare database

	Set A	Set B
Unique users	704	614
Total reviews	4477	2925

3. Convert the discovered centre points into the original dimension of reviews, that is, the words.
4. The most heavily occurring or weighted term in each centre describe a particular topic.

All the topics were distributed evenly throughout the dataset. The results of the *k*-means clustering process is shown in Figure 7.

4. EVALUATION

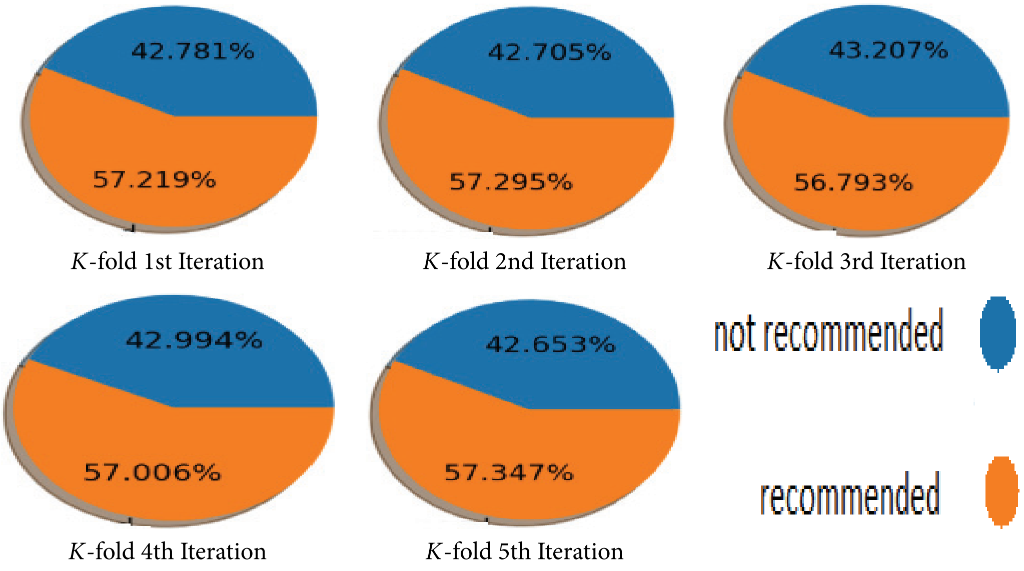
We employ an offline evaluation approach for the LBRS. We used the processed datasets discussed in 3.1, Set A and Set B, to evaluate our model. Table 3 shows the unique users and total reviews for each Set.

Following the guidance of Salzberg [28], data is stratified before being splitting it into k parts. Any machine learning model can only be evaluated purposefully if it is tested on unseen data. This assists us in determining if it is under-fitting, over-fitting or well-generalised. Forming test and training data through appropriate division is often not considered by researchers, and inappropriate data-splitting may lead to high difference in model performance, largely through inaccuracy. Thus, we implemented *k*-Fold cross validation and also injected our Set B dataset into the training set due to the sparsity of our actual training dataset to avoid the over-fitting and under-fitting problems. To apply correct data division ration for training and testing, we will apply the *k*-fold technique to select test and training data from the Set A for 'k' number of times. This technique imitates the usage of test and training sets by continuously training the algorithm *k* number of times.

Review data was unlabelled, so we labelled text using the emotions generated. For sentiment analysis, we passed the data to the SentiStrength model to calculate the sentiment score and label the data [32].

4.1. Baseline Models

We evaluated the proposed recommendation system that uses both topic modelling and sentiment analysis by comparing it against the performance of models built from only sentiment analysis. To measure the overall accuracy, we use the baseline models SVM, Random Forest, Logistic Regression and Naive Bayes for testing the robustness and efficiency of our new classification model. These models are selected because they are widely accepted, consistent and reliable. They were built using only sentiment score as ratings, and user and venue data as feature vectors.



K-fold 1st Iteration

K-fold 2nd Iteration

K-fold 3rd Iteration

K-fold 4th Iteration

K-fold 5th Iteration

Figure 8 | *k*-fold even distribution of data in each fold.

4.2. Performance Measures

We used a confusion matrix to to evaluate the performance of our intended models. The following criteria was used to conduct the performance measures for our experiment:

Accuracy is a ratio of correctly anticipated outcomes to the total outcomes which help us to determine how often the classifier is correct [30].

$$\text{Accuracy} = \frac{TN + TP}{TN + FP + FN + TP} \tag{4.1}$$

Precision is the ratio of accurately anticipated positive outcomes to the total predicted positive outcomes [30].

$$\text{Precision} = \frac{TP}{FP + TP} \tag{4.2}$$

Recall is the ratio of accurately anticipated true observations to the all observations in the real scenario [30].

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

F1 Measure is weighted mean of recall and precision [30].

$$F \text{ measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.3}$$

4.3. Experiment

Porter Stemmer was selected for our experiment, as it is widely accepted in the discipline [10]. The dataset is divided into 5(k) nearly equal folds or segments. In these partitioned sections, training and testing is done 5 times and, in each iteration, we leave one segment for testing and train the model on the remaining 4 folds. In our case, we also merged Set B with these 4 folds to eliminate the data sparsity problem. The training set is used to train the classifier, and the test set is used to measure the performance of the predictive model. The results of the *k*-means evaluation is shown in Figure 8.

Table 4 | Mean score after *k*-fold classification with only sentiment

Classifiers	Accuracy	F1 score	Precision	Recall
Naive bayes	0.501	0.65	0.497	0.975
Logistic regression	0.529	0.483	0.527%	0.457
SVM	0.559	0.318	0.87	0.307
Random forest	0.616	0.53	0.619	0.550

Table 5 | Performance of various classifiers

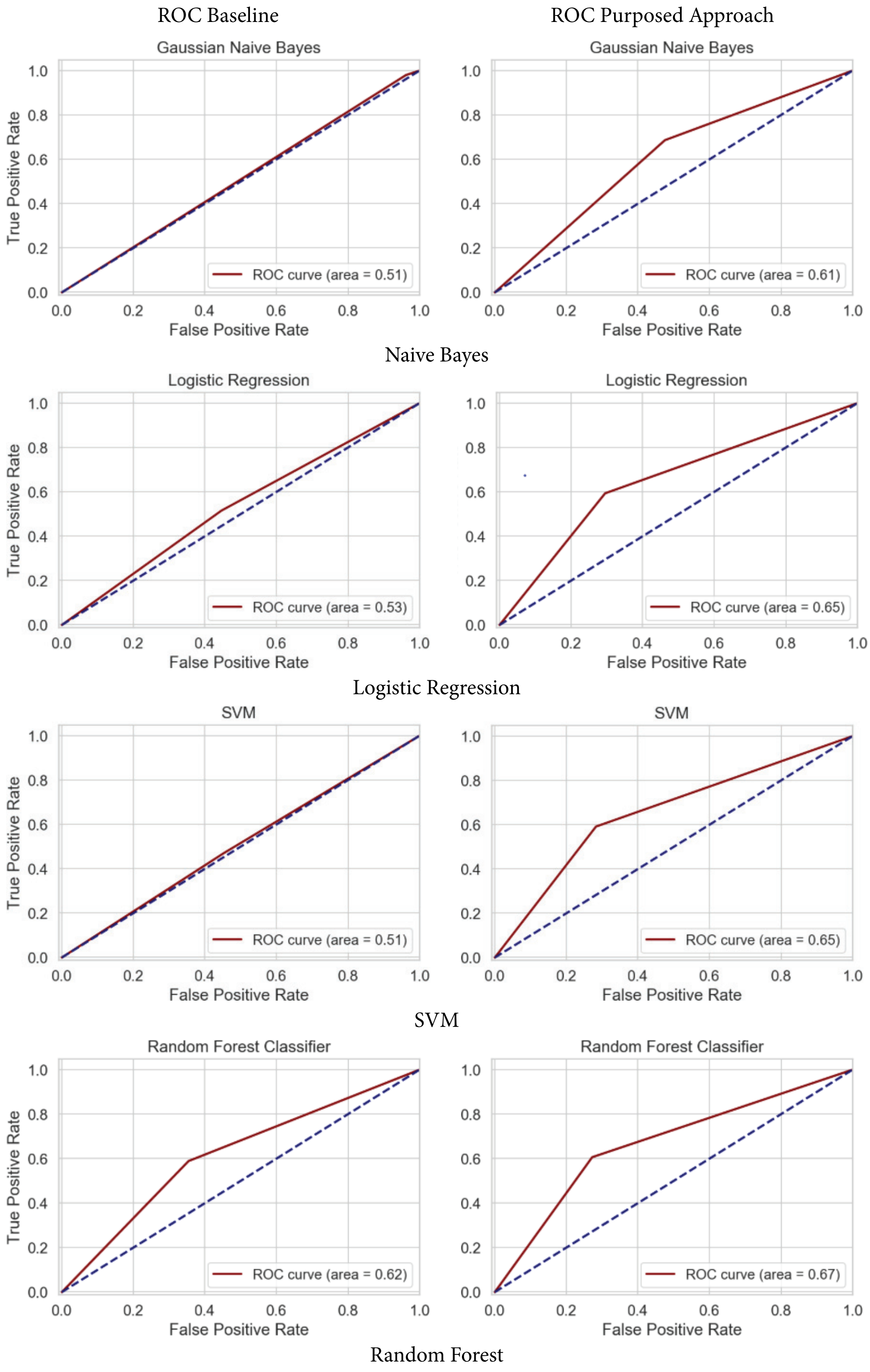
Classifiers	Accuracy	F1 score	Precision	Recall %
Naive bayes	0.606	0.63	0.58	0.686
Logistic regression	0.649	0.625	0.664	0.59
SVM	0.665	0.628	0.670	0.591
Random forest	0.675	0.648	0.686	0.608

We firstly calculated the performance of different models that used only sentiment analysis. The accuracy, F1 score, precision and recall obtained in each iteration is averaged to get the model mean performance score for each metric. The results are shown in Table 4.

Random Forest performed the best for our baseline model, with 0.616 accuracy score. The F1 score of Naive Bayes is higher, but it is significantly influenced by the higher recall score. In our purposed LBRS, we focus more on precision than on recall, so despite the F1 score being high for Naive Bayes, the Precision is low compared to Random Forest. SVM has the second-best accuracy score of 0.559 and has a very good precision score of 0.87. Logistic Regression had an average performance for all the metrics.

In the second iteration of the experiment, the purposed LBRS is implemented using combined sentiment analysis and topic modelling. These results are shown in Table 5.

As seen in Table 5, Random Forest performs well for our baseline model, with an accuracy score of 0.675, higher than any other baseline model. The weighted average of precision and recall i.e. the F1 score is also highest at 0.648. All the other classifiers performed well with our approach, producing an accuracy score of more than 0.6, which is very good for a classification model.



ROC Baseline

ROC Purposed Approach

Naive Bayes

Logistic Regression

SVM

Random Forest

Figure 9 | ROC curves.

For performance efficiency, we use an ROC curve to plot the true positive rate (TPR) against the false positive rate (FPR) from our results. The graphs are compared in Figure 9. In the ROC graph for the baseline models, area under curve (AUC) is highest for Random Forest. Next is Logistic Regression, with a slightly better curve than the other two models, SVM and Naive Bayes. For the purposed method, all of the ROC curves were found to perform higher. But with the highest AUC, Random forest thus proved to be the best classification model using our approach.

We found that using sentiment score alone produced disappointing results. The low performance of sentiment in determining venue recommendations indicates that user profiles require more specific features, like interests and topics. Each classification model received a boost in performance during the experiment after the addition of topic modelling features. The highest accuracy for the hybrid model of 0.67 was obtained by Random Forest, with a 5% increase in accuracy and 9% increase in F1 score compared to its results when considering only sentiment in the training data. Precision and recall have a significant difference, meaning that our model gives more priority to a label. Especially, the Random Forest and SVM models outperformed the other models and their performance was improved in all areas compared to the baseline approaches.

In Naive Bayes, only accuracy and precision were found to be better than its baseline counterpart. However, this is not significant because in an RS, precision deals with not making any wrong recommendation whereas recall refers to not missing the right answers. Among the tested models, Random Forest has the best performance in both scenarios and was more accurate and stable than the other models overall. Compared to the models built only from sentiment analysis, our purposed LBRS shows significant improvement in all criteria of the evaluation metrics.

We can conclude that the performance of the combined model was higher than the models using sentiment analysis only. Among the various models, ROC is superior for the Random Forest classifier model, which is also the acknowledged model from evaluation metrics as well. Our results demonstrate that sentiment must be fused with topic classification for better user profiling, and tools for its analysis should be designed to generate better recommendation models. Additionally, during our experiment, we found that splitting the dataset proved a better way to balance the *k*-fold iterations.

5. CONCLUSION

Our proposed technique is a hybrid model implementing SentiStrength for Sentiment analysis and LDA for topic modelling to create a user profile that gives the best recommendation possible for a POI. Location reviews were collected from Foursquare, with the data labelled through sentiment analysis and fused with topic modelling to profile users according to their posting behaviour. We then used several classifiers to create a recommendation model that suggests best venues to the users. Compared to previous work that only considered user similarities, our proposed method focuses on mining semantic information like features, topics and sentiments associated with venues to generate the POI recommendation. Embedding topics is more effective than the baseline models that use sentiment score alone, as overall accuracy improved from 0.61 to 0.67 (based on our best classifier outcome using Random Forest).

This provides important insight into hybrid data mining techniques, as a combined method generates more accurate predictions. Check-in data only ranks the location as positive and negative without considering a user's actual emotions regarding the venue, but with Sentiment analysis, we can determine a visitor's emotion towards the venue. Practically, this enables future designers to consider review data and extract semantic features beyond check-ins when developing RS. The upshot of this research project work is the ability to group reviews in a significant way and to detect the user sentiments on the underlying category of reviews.

The limitations of this research were that we only extracted user tips from the dataset for user modelling. Future work could extend this by considering tags, geography and other features for the user profile model, which would improve recommendation effectiveness. It might be useful to profile user models according to the frequency of their visit and reviews, user similarities based on their social circle, and the propagation of their network. We could also explore the impact of users with broader travel histories and locations visited, compared to users who don't travel as often. Considering further features may also help filter bots or spam in the reviews.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

AUTHORS' CONTRIBUTION

Xiaohui Tao contributed in Conceptualization, Methodology, Validation, Formal analysis, Investigation, Writing. Nischal Sharma contributed in Methodology, Software, Validation, Formal analysis, Investigation, Writing. Patrick Delaney contributed in Conceptualization, Methodology, Formal analysis, Writing. Aimin Hu contributed in Conceptualization, Methodology. All authors reviewed the results and approved the final version of the manuscript.

REFERENCES

[1] A.M. Abirami, V. Gayathri, A survey on sentiment analysis methods and approach, 2016 Eighth International Conference on Advanced Computing (ICoAC), IEEE, Chennai, India, 2017, pp. 72–76.

[2] S. Aciar, D. Zhang, S. Simoff, J. Debenham, Recommender system based on consumer product reviews, Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence, IEEE, Hong Kong, China, 2006.

[3] E. Çano, M. Morisio, Hybrid recommender systems: a systematic literature review, Intelligent Data Analysis 21 (2017), 1487–1524.

[4] P.B. Jitha, M. Reshma, Literature survey on tourism recommendation, International Journal of Innovative Research in Computer and Communication Engineering 5 (2017), 1916–1922.

[5] J. Bao, Y. Zheng, D. Wilkie, M. Mokbel, Recommendations in location-based social networks: a survey, Geoinformatica 19 (2015), 525–565.

[6] B. Berjani, T. Strufe, A recommendation system for spots in location-based online social networks, Proceedings of the 4th workshop on social network systems, ACM, New York, United States, 2011, pp. 1–6.

[7] P. Chavan, Analytical study on collaborative filtering techniques for location-based recommendation, International Journal of Scientific & Engineering Research 6 (2015), 1750–1758.

[8] Z. Cheng, J. Caverlee, K. Lee, D. Sui, Exploring millions of footprints in location sharing services, Proceedings of the International AAAI Conference on Web and Social Media (ICWSM), 2011, pp. 81–88. Retrieved from: https://ojs.aaai.org/index.php/ICWSM/article/view/14109.

[9] Y. Doytsher, B. Galon, Y. Kanza, Storing routes in socio-spatial networks and supporting social-based route recommendation, Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location-Based Social Networks (LBSN '11), ACM, New York, NY, USA, 2011, pp. 49–56.

[10] M. Dumanskyi, Detection of text after neural network's classification, Prospects for the Development of Modern Science and Practice, Abstracts of XVI International Scientific and Practical Conference, Graz, Austria, 2020, pp. 32–33.

[11] N. Ganganath, C.-T. Cheng, C.K. Tse, Data clustering with cluster size constraints using a modified K-means algorithm, 2014 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, IEEE, Shanghai, China, 2014, pp. 158–161.

[12] Z. Guan, T. Ji, X. Qian, Y. Ma, X. Hong, A survey on big data pre-processing, 2017 5th Intl Conf on Applied Computing and Information Technology/4th Intl Conf on Computational Science/Intelligence and Applied Informatics/2nd Intl Conf on Big Data, Cloud Computing, Data Science (ACIT-CSII-BCD), IEEE, Hamamatsu, Japan, 2017, pp. 241–247.

[13] F.O. Isinkaye, Y.O. Folajimi, B.A. Ojokoh, Recommendation systems: principles, methods and evaluation, Egyptian Informatics Journal 16 (2015), 261–273.

[14] H. Kautz, B. Selman, M. Shah, Referralweb: combining social networks and collaborative filtering, communications of the ACM 40 (1997), 63–65.

[15] S. Kim, How foursquare and other apps guess what you want to eat, 2015, available from: https://www.eater.com/2015/4/24/8486279/restaurant-recommendation-apps-foursquare-algorithms.

[16] C. Leung, S. Chan, F.L. Chung, Integrating collaborative filtering and sentiment analysis: a rating inference approach, Proceedings of the ECAI 2006 workshop on recommender systems, 2006, pp. 62–66.

[17] D. Lian, Y. Ge, F. Zhang, N.J. Yuan, X. Xie, T. Zhou, et al., Content-aware collaborative filtering for location recommendation based on human mobility data, Proceedings of the 2015 IEEE International Conference on Data Mining (ICDM), IEEE Computer Society, Washington, DC, USA, 2015, pp. 261–270.

[18] L. Huang, Q. Li, Y. Yue, Activity identification from GPS trajectories using spatial temporal POIs' attractiveness, Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks, (LBSN '10), ACM, New York, NY, USA, 2010, pp. 27–30.

[19] G. Linden, B. Smith, J. York, Amazon.com recommendations: item-to-item collaborative filtering, IEEE Internet Computing 7 (2003), 76–80.

[20] F. Liu, H.J. Lee, Use of social network information to enhance collaborative filtering performance, Expert Systems with Applications 37 (2010), 4772–4778.

[21] K. Madadipouya, S. Chelliah, A literature review on recommender systems algorithms, techniques and evaluations, Broad Research in Artificial Intelligence and Neuroscience 8 (2017), 109–124.

[22] I. Mogotsi, Christopher d. manning, prabhakar raghavan, and hinrich schütze: introduction to information retrieval, Information Retrieval 13 (2010), 192–195.

[23] K. Nur'aini, I. Najahaty, L. Hidayati, H. Murfi, S. Nurrohmah, Combination of singular value decomposition and K-means clustering methods for topic detection on Twitter, 2015 International Conference on Advanced Computer Science and Information Systems (ICACSIS), IEEE, Depok, Indonesia, 2015, pp. 123–128.

[24] B. Pang, L. Lee, Opinion mining and sentiment analysis, Foundations and Trends in Information Retrieval 2 (2008), 1–135.

[25] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, et al., Scikit-learn: machine learning in Python, Journal of Machine Learning Research 12 (2011), 2825–2830.

[26] M. Špilka, A. Posoldova, G. Rozinaj, P. Podhradský, Importance of recommendation system in modern forms of learning, 2016 International Conference on Systems, Signals and Image Processing (IWSSIP), IEEE, Bratislava, Slovakia, 2016, pp. 1–4.

[27] S. Gigimol, S. John, A survey on different types of recommendation systems, International Research Journal of Advanced Engineering and Science 1 (2016), 111–113.

[28] S.L. Salzberg, On comparing classifiers: pitfalls to avoid and a recommended approach, Data Mining and Knowledge Discovery 1 (1997), 317–328.

[29] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, Item-based collaborative filtering recommendation algorithms, Proceedings of the 10th International Conference on World Wide Web, WWW '01, ACM, New York, NY, USA, 2001, pp. 285–295.

[30] B. Seref, E. Bostanci, Sentiment analysis using naive bayes and complement naive bayes classifier algorithms on hadoop framework, 2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), IEEE, Ankara, Turkey, 2018, pp. 1–7.

[31] M. Taboada, J. Brooke, M. Tofiloski, K.D. Voll, M. Stede, Lexicon-based methods for sentiment analysis, Computational Linguistics 37 (2011), 267–307.

[32] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, A. Kappas, Sentiment strength detection in short informal text, Journal of the American Society for Information Science and Technology 61 (2010), 2544–2558.

[33] H. Wang, M. Terrovitis, N. Mamoulis, Location recommendation in location-based social networks using user check-in data, Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM, New York, NY, USA, 2013, pp. 374–383.

[34] H. Werthner, A. Alzua-Sorzabal, L. Cantoni, A. Dickinger, U. Gretzel, D. Jannach, et al., Future research issues in IT and tourism: a manifesto as a result of the JITT workshop in June 2014, Vienna, Information Technology & Tourism 15 (2015), 1–15.

[35] X. Xiao, Y. Zheng, Q. Luo, X. Xie, Finding similar users using category-based location history, Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM, New York, NY, USA, 2010, pp. 442–445.

[36] D. Yang, D. Zhang, Z. Yu, Z. Wang, A sentiment-enhanced personalized location recommendation system, Proceedings of the 24th ACM Conference on Hypertext and Social Media, ACM, New York, NY, USA, 2013, pp. 119–128.

[37] D. Yang, D. Zhang, Z. Yu, Z. Yu, Fine-grained preference-aware location search leveraging crowdsourced digital footprints from LBSNs, Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing, ACM, New York, United States, 2013, pp. 479–488.

[38] M. Ye, P. Yin, W.-C. Lee, Location recommendation for location-based social networks, Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM, New York, NY, USA, 2010, pp. 458–461.

[39] M. Ye, P. Yin, W.-C. Lee, D.-L. Lee, Exploiting geographical influence for collaborative point-of-interest recommendation, Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '11, ACM, New York, NY, USA, 2011, pp. 325–334.

[40] Y. Zheng, L. Zhang, X. Xie, W.-Y. Ma, Mining interesting locations and travel sequences from GPS trajectories, Proceedings of the 18th International Conference on World Wide Web, ACM, New York, NY, USA, 2009, pp. 791–800.

[41] R. Zachery, S.-K. Wang, An explanation of linear discrete time system behavior by singular value decomposition of the system matrix, Proceedings of 28th Southeastern Symposium on System Theory, IEEE, Baton Rouge, LA, USA, 1996, pp. 18–22.

[42] J.-D. Zhang, C.-Y. Chow, Y. Zheng, ORec: an opinion-based point-of-interest recommendation framework, Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, ACM, New York, NY, USA, 2015, pp. 1641–1650.

[43] V.W. Zheng, Y. Zheng, Q. Yang, Joint learning user's activities and profiles from GPS data, Proceedings of the 2009 International Workshop on Location Based Social Networks, ACM, New York, NY, USA, 2009, pp. 17–20.

[44] Y. Zheng, X. Xie, Learning location correlation from GPS trajectories, 2010 Eleventh International Conference on Mobile Data Management, IEEE, Kansas City, MO, USA, 2010, pp. 27–32.