Fayoum University

Contents lists available at Egyptian Knowledge Bank

Labyrinth: Fayoum Journal of Science and Interdisciplinary Studies

Journal homepage: https://lfjsis.journals.ekb.eg/

Labyrinth Journal

Food Interests Analysis (FIA) model to extract the food preferences and interests of Twitter users

Abdalla M. Mohamed [a,*], Haytham Al-Feel [a, b], Shereen A. Taie [a]

[a] Department of Information Systems, Faculty of Computers and Artificial Intelligence, Fayoum University, El Fayoum 63514, Egypt
[b] Department of Computer Science, Community College, Imam AbdulRahman Bin Faisal University, Dammam 31441, Saudi Arabia

ARTICLE INFO

Keywords:
Topic modeling
Streaming data
Sentiment analysis
Business intelligence
Twitter
food preferences

ABSTRACT

Online social networks like Facebook and Twitter have played an important role in networking, disseminating information, and sharing interests and entertainment since the internet's advent into our daily lives. Twitter has significantly contributed to the analysis of its user-generated data for personalization and tasks of recommendation due to its rapid growth as a social networking platform. Twitter posts serve as an important source of information for identifying users' positive interests and creating intelligent recommendation systems. These posts provide a lot of information that may be analyzed to determine users' preferences on various topics, including food. Twitter post analysis is an interesting field of study. Several studies have studied the sentiment analysis of tweets. Also, market forecasting is a crucial issue that requires careful consideration. Business intelligence (BI) becomes an important analytical technique for assessing consumer satisfaction and market demand. Since business intelligence requires in-depth analysis, sentiment analysis is the process of using natural language processing (NLP) and machine learning (ML) techniques to identify the emotional tone and attitude in text, making it useful for analyzing Twitter posts and customer reviews and identifying user preferences and market demand. As a result, it's critical to choose relevant advertisements for users at particular locations to capture their attention and generate profit. This paper develops a proposed model for a Food Interests Analysis (FIA). It collects 20,000 publicly available tweets, and then the sentiments conveyed in the tweets are captured and normalized, then clustered according to the common topic. This paper also examines the accuracy of two lexicon-based sentiment analysis approaches for tweets. Also, this study proposes an approach that combines both topic modeling and sentiment analysis (SA) by Latent Dirichlet Allocation (LDA) using the term frequency-inverse document frequency (TF-IDF) and extracting sentiment from tweets. Thus, this approach can identify the food preference categories in which users are interested.

1. Introduction

In recent years, the number of users of online social networks has increased rapidly [1]. With the growth of the Internet, blogs, and mobile devices, social media has also become an information provider through the publishing and sharing of content generated by users. The amount of text data available on the web and computers is rapidly increasing, so handling that text data requires intelligent algorithms to retrieve useful information from data repositories [2]. The analysis of textual data, which contains the thoughts and communication among users, allows for an understanding of the public's requirements and concerns about what defines valuable information from academic and marketing perspectives [3].

Twitter (http://twitter.com) is one of the popular social media networks that allows its users to create and obtain valuable information about reviews, interests, issues, and trends in real time using short texts called tweets [3]. These tweets can be used to analyze user interests and identify trends going on anywhere. With the use of such analysis, a smart recommendation system may be created. The field of online social networks has produced several studies, including those on the classification of topics, gender, sentiment analysis of Twitter users based on tweets, community detection, event detection, etc.

Most research on the recommendation system is based on network topology. If additional information, such as demographic characteristics, a user's interests, and the interests of other users, is considered, a user's knowledge of social media sites might be significantly enhanced. With this knowledge, users may follow posts or people based on the topics they are interested in, and they can also join communities that interest them. Also, a user can be interested in receiving recommendations depending on her areas of interest, such as food preferences.

[*] Corresponding author.

E-mail address: amm22@fayoum.edu.eg (A. M. Mohamed); Tel.: +201069833661

DOI: 10.21608/IFJSIS.2023.204964.1010
Received 10 April 2023; Received in revised form 14 May 2023; Accepted 25 May 2023
Available online 26 May 2023

As mentioned above, nowadays, social networks contain a huge amount of users' tweets. Today, a huge number of users' tweets are available on social networks. As a result, studying them may help us identify users' preferences in those fields. These tweets contain a lot of information in a variety of fields, like commerce, education, and tourism. Since tweets are written using natural languages [4], NLP techniques should be applied to process them.

The extraction of user preferences is an issue with many applications on the Internet of Things (IoT). For example, it may be used as input information for personalized and context-aware IoT applications and recommender systems [5]. The contextual information collected by IoT sensors and devices helps these systems adjust their recommendations to the user's situation [6].

Since many social media users suffer from information overload, finding useful content in huge text streams is a critical challenge. Food is considered one of the most basic human needs. Many people place a great deal of importance on eating their desired foods. In this regard, discovering a user's food preferences from Twitter posts could be useful for various applications, such as those that try to find restaurants that fit the user's taste. Also, applications that use artificial intelligence (AI) for restaurant advertisements can fetch appropriate users already interested in the categories of food presented by these restaurants.

In this research work, the tailored suggestions have been improved depending on the user's preferences by filtering information from Twitter text. Also, the goal of this paper is to extract users' food preferences by analyzing their tweets. After accurately clustering tweets based on common topics and sentiment analysis, the model constructs user food interests by analyzing user tweets to discover food user preferences. The rest of the paper is organized as follows: methods in Section 2 that explains the reasons for using certain techniques or approaches to identify, select, and analyze the information applied and also presents the steps for the proposed methods and their components. Followed by results and discussion in Section 3, which covers the implementation technique and presents the evaluation results of the proposed methods. In the end, Section 4 includes a conclusion summarizing the research work and indicating future work.

2. Related work

Due to the popularity, accessibility, and variety of fast food, fast food restaurants have become prevalent in all types of societies. These restaurants' reviews have been subjected to sentiment analysis and topic modeling with machine learning and deep learning models.

For the study of customer reviews, surveys, and tweets, sentiment analysis is the most essential and widely employed technique in the modern era. Researchers from all over the world have been engaged in sentiment analysis research. Text mining and a lexicon-based approach were used to extract a set of Twitter-based sentiments (positive or negative) about food delivery [7]. In 2021, Ahmed et al. [8], online food evaluations are analyzed using a big data-based approach. The Apache Spark system uses support vector classifiers (SVC), logistic regression, and Naive Bayes to analyze the dataset for Amzaon's fine cuisine reviews. The results indicate that SVC outperforms other models. Consider the online food delivery applications Swiggy, Zomato, and UberEats to analyze sentiment using lexicon-based word emotion and lexicon-based sentiment classification [7]. Various findings regarding positive and negative remarks regarding the selection of online applications are discussed. The tweets relevant to market information were preprocessed using text-mining and text-preprocessing techniques, and sentiments were assigned using the TextBlob method to enhance the performance of lexicon sentiments [9]. Using the Tweepy Python library, Pokharel [10] collected 615 messages from Twitter. The authors removed links, spaces, punctuation, and stopwords from the tweets and categorized them as positive, negative, or neutral using the TextBlob approach.

In 2022, Wahyuni [11] explored the implementation of the Support Vector Machine (SVM) method for sentiment analysis using Twitter data. The study involves pre-processing the data by removing stop words, stemming, and tokenization. The SVM algorithm is then applied to classify tweets into either positive, negative, or neutral sentiments. The results show that the SVM method can achieve an accuracy of 80% for sentiment analysis of Twitter data, which is a promising outcome for future research in this area.

In 2020, Prananda and Thalib [12] a case study on analyzing user reviews for the Indonesian ride-hailing service GO-JEK using sentiment analysis is presented. A dataset of customer reviews collected from social media platforms and online forums was evaluated by sentiment analysis using a Naive Bayes classifier. The study evaluates the performance of the system using accuracy, precision, recall, and F1 score metrics and shows that the system achieves an accuracy of 83.4%, a precision of 85.9%, a recall of 81.2%, and an F1 score of 83.5%. The study suggests that sentiment analysis can provide valuable insights for companies like GO-JEK to improve their services and enhance customer satisfaction.

Khattak et al. [13] proposed a personalized tweet recommendation system based on sentiment analysis and classification. The study uses a dataset of tweets collected from Twitter and applies natural language processing techniques to identify the sentiment of the tweets and classify them into categories. Also, use various classification algorithms to classify the tweets and recommend personalized tweets to users based on their interests. The results show that the proposed system is effective in recommending personalized tweets to users, with the Naive Bayes classifier achieving an accuracy of 85.47%. The study highlights the potential of sentiment analysis and classification in personalized tweet recommendation systems. In 2022, Hadi et al. [14] presented a study on the sentiment analysis of Go-Food using Twitter data. The study compares the performance of the Random Forest algorithm and the Linear Support Vector Classifier in classifying tweets as positive, negative, or neutral. The results show that both algorithms performed well in classifying the tweets, with the Random Forest algorithm achieving a slightly higher accuracy of 83.4% compared to the Linear Support Vector Classifier's accuracy of 81.4%. The study provides insights into customers' opinions and sentiments towards Go-Food using Twitter data and shows the effectiveness of both algorithms in sentiment analysis.

Topic modeling has introduced new revolutions to the areas of text mining and sentiment analysis. It is an approach for discovering latent meaning in text datasets by mining statistical patterns. Several scholars have published articles on topic modeling. In Maier et al. [15] a group of researchers introduces LDA topic modeling as a valid and reliable methodology for communication research. They discuss the processes of data preprocessing, parameter estimation, and interpretation of results for the LDA topic modeling method. The authors conclude that LDA topic modeling is an effective method for communication research and provide guidelines for its use in future studies. The research offers valuable insights into the application of LDA topic modeling in communication research and its potential to uncover patterns in large sets of data. Naskar et al. [16] analyzed user sentiments by collecting messages from social networking sites, particularly Twitter. The sentiments include emotions and diverse topics that are categorized using the LDA methodology. In another study, Jelodar et al. [17] examined various LDA-based approaches for identifying research development as well as the applications and challenges associated with these methods. This paper proposes a method for detecting user interests from Twitter data using semantic analysis techniques.

Zarrinkalam et al. [18] used LDA and WordNet to extract topics from Twitter data and compute the semantic similarity between user interests and the extracted topics. The authors evaluate the system's performance using a classification model, and the results show that the proposed method achieves an accuracy of 73.6% in detecting user interests. The study suggests that using semantic analysis techniques can improve the accuracy of detecting user interests from Twitter data, and the proposed method has potential applications in targeted marketing and personalized content recommendation. In 2016, [19] analyzed users' interests based on their tweets using LDA to model the topics. Tweets from Twitter are collected, preprocessed, and their topics of interest identified, which include sports, entertainment, politics, technology, and education. The study observed that users tend to tweet more about specific topics of interest and that there is a correlation between the number of tweets and the level of interest in a particular topic. It provides insights into users' interests and can be useful in personalized recommendation systems.

A topic modeling approach using LDA is proposed in [20] to extract topics from a dataset of customer reviews of hotels and restaurants. The system's performance is evaluated using perplexity and coherence metrics and compared with a human-labeled set of topics. The proposed method effectively extracts relevant topics from the reviews and achieves a high level of agreement with the human-labeled set, as shown by the study. Valuable insights into customer feedback can be provided by topic modeling, according to the results, enabling businesses to improve their products and services based on customer preferences. The research described in 2020 [21] utilized machine learning algorithms to predict user interests based on Urdu tweets. A large dataset of tweets is used, and text preprocessing techniques, feature extraction, and classification algorithms are applied. The performance of different classifiers, including SVM, KNN, Random Forest, and Naïve Bayes, is evaluated using accuracy, precision, recall, and F1 score metrics. The Random Forest classifier is found to perform the best, achieving an accuracy of 81.54%, precision of 81.55%, recall of 81.54%, and F1 score of 81.54%. It is concluded that predicting user interests based on Urdu tweets can have important implications for targeted advertising, personalized recommendations, and other applications. Milani et al. [22] presented a sentiment analysis approach to extract and classify users' interests from tweets. The proposed approach combines a lexicon-based approach with machine learning algorithms to extract and classify the sentiment of tweets related to different topics. The approach is evaluated on a dataset of 16,000 tweets and achieves an accuracy of 80% for sentiment extraction and 72% for topic classification. The study demonstrates the potential of sentiment analysis for extracting valuable insights from social media data for various applications, including marketing and customer feedback analysis.

In 2020, Asani et al. [23] presented a novel approach to extracting the food preferences of social media users using sentiment analysis. The authors collect a large dataset of food-related tweets, preprocess the data, and classify the tweets into positive, negative, and neutral sentiments. They then apply association rule mining to identify frequent food items that co-occur with positive or negative sentiments. The study shows that sentiment analysis can effectively extract the food preferences of users, enabling personalized food recommendations and improved customer satisfaction in the food industry. The results suggest that sentiment analysis can be a powerful tool for analyzing social media data to gain valuable insights into customer preferences.

3. Materials and Methods

In this portion, the proposed model FIA extracts users' food preferences to understand the food interests of Twitter users. The processes of the FIA proposed model consist of some steps, where the first step is sentiment analysis, which involves automatically classifying and quantifying the sentiment expressed in textual data that includes the collection of tweets using the Twitter API. Since the dataset contains a vast amount of data with various words and slang, the specific tweets are analyzed and classified using text analytics and NLP. Then, preprocessing is applied. In this stage, the collected tweet data is then analyzed to remove unwanted words, mentions, URLs, punctuation, extra spaces, and emoticons, in addition to the duplicate records, to extract important information [24, 25]. Consequently, breaking down the cleaned text data into smaller units called tokens. Tokenization is usually performed using whitespace or punctuation as delimiters. After that, convert all the tokens to lowercase to ensure that words with different capitalizations are treated as the same token. Then removing stop words from the tokenized text data to reduce the size of the vocabulary and improve the efficiency of subsequent NLP tasks. Finally, avoid duplicating the same word by normalizing the tokens in the text data by reducing them to their root or base form; this is called the stemming and lemmatization process.

In the second step, classification using machine learning algorithms, which involves training algorithms on labeled data to accurately predict the class of unseen data points, is applied.

In the third step, a topic modeling algorithm is applied using LDA to retrieve information and select features from unstructured text in the context of information retrieval. As a topic modeling algorithm, LDA is valuable in organizing extensive amounts of textual data into intersecting clusters of documents [26]. LDA will group all tweets that relate to each other around a common topic to group tweets that relate to food in a certain topic and another topic for non-related food topics. In the fourth stage, apply SA, which is a natural language processing technique that involves analyzing and classifying the subjective content of a text into positive, negative, or neutral sentiments to extract user preferences.

This research is carried out using some of the processes for FIA proposed FIA model shown in Fig. 1. The next subsection describes each of these steps in detail.

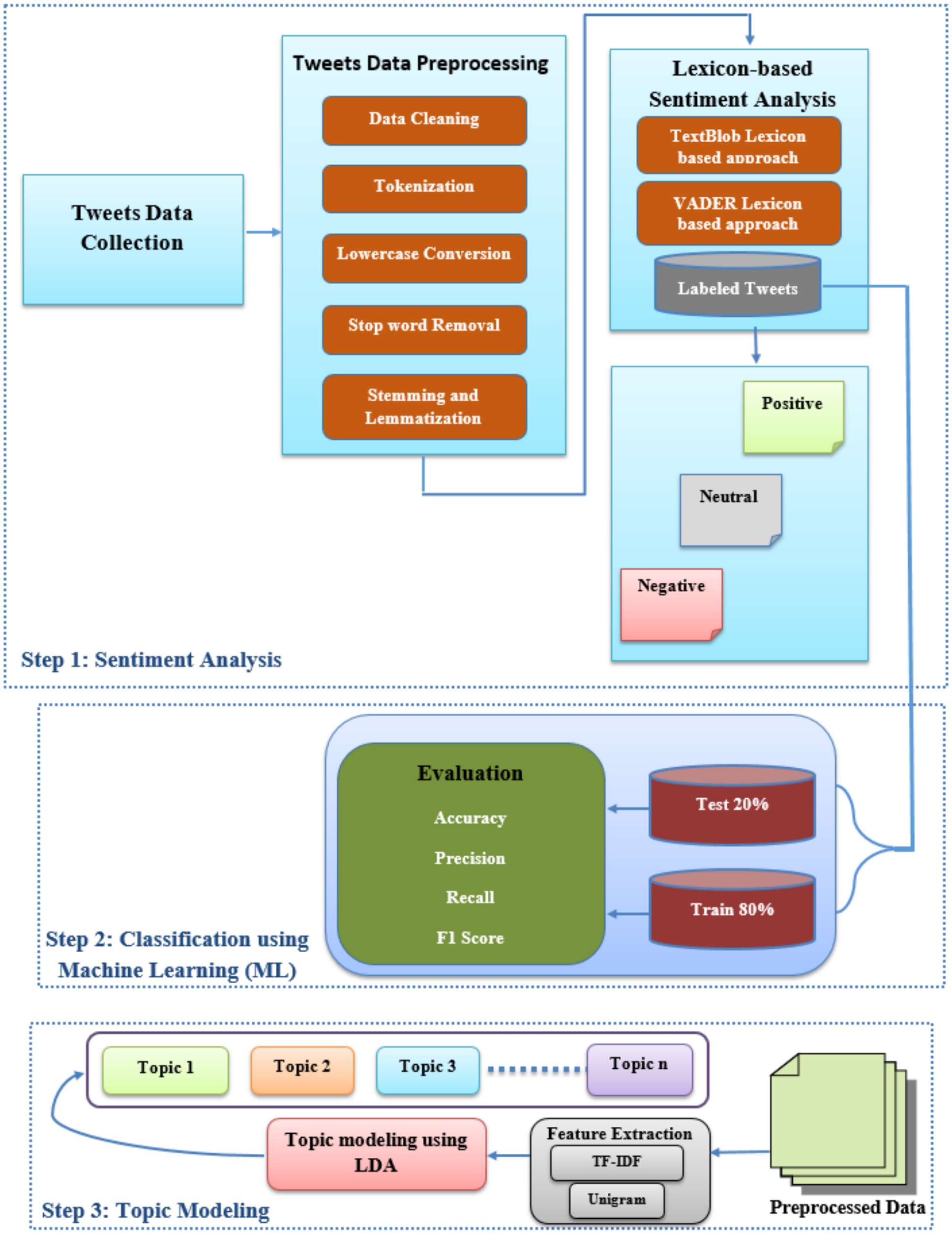


Fig. 1. Workflow diagram of the FIA model methodology for sentiment analysis and topic modeling.

3.1. Data collection

Data collection is an essential first step in NLP tasks. In this stage, data is gathered from various sources and loaded into a suitable format for analysis. One common source of data for NLP is social media platforms, such as Twitter.

To collect data from Twitter, the Snscrape Python library is used to fetch 20,000 tweets along with important information such as the User ID, User Name, Date, Tweet, and Location.

The dataset contains the following columns:

- User ID: The unique identifier of the Twitter user who posted the tweet
- User Name: The username of the Twitter user who posted the tweet

- Date: The date when the tweet was posted
- Tweet: The text content of the tweet
- Location: The location where the tweet was posted (e.g. Canada, Maharashtra, India, etc.).

Snscrape is a powerful and easy-to-use library that allows scraping data from Twitter by simply specifying the search query and the number of tweets required. In this study, 20,000 tweets were collected to ensure that there was enough data to perform a meaningful analysis. Once the data was collected, it was loaded into a suitable format for analysis, which could involve converting the data into a structured format.

3.2. Preprocessing

Data obtained from diverse sources contains both useful and useless information. Therefore, the data needed to be clear before further processing [21]. So before applying any machine learning algorithms, data preparation is a crucial phase. Text data requires data preparation before applying any machine learning algorithm. Data preprocessing is done to prepare the data [2]. This study investigates if the preprocessing phase has any effect on the quality of the results. In the preprocessing step, it is usually decided how to handle the data: by cleaning, normalizing, or leaving it unchanged.

The model's performance is enhanced by performing major preprocessing steps. Preprocessing is an important stage in NLP tasks. It converts input text into a more desirable form to perform better in subsequent processes [27]. The role of the pre-processing stage is crucial to the text clustering and classification processes. Preprocessing of text involves cleaning up noise such as stop words, punctuation, mentions, URLs, extra spaces, emoticons, and terms that don't carry much weight in the context of the text, such as words of two characters, etc. The data that collected from Twitter comes from a variety of sources. As the dataset includes various types of data from Twitter, it is necessary to take certain measures to prepare and normalize the data.

3.2.1. Data cleaning

Data cleaning is crucial for natural language processing (NLP) tasks. It ensures the analyzed data is accurate, relevant, and free of unwanted noise or errors. For example, in our case of collecting publicly available tweets, data cleaning was an essential step to ensure that the data being analyzed was of high quality and meaningful. This stage was used to perform a data cleaning step. The Twitter cleaning procedure was carried out in the first phase of the preprocessing stage. We refine the tweet by removing unwanted words, mentions, URLs, punctuation, extra spaces, emoticons, and terms that don't carry much weight in the context of the text, such as words of only two characters that don't add any value to the text data, to make it more readable as plain text.

In addition, remove the duplicate records to extract important information. Finally, this step ensures that the data being analyzed is of high quality and meaningful, which in turn helps extract valuable insights and knowledge from the text data.

For example, tweet is "I love to eat grilled #chicken when I get back home tired; just grilled chicken is all that can make me #happy.", so tokens will be:

"I love to eat grilled chicken when I get back home tired just grilled chicken is all that can make me happy".

3.2.2. Tokenization

Natural language text data must be converted into a readable format that can be used by machine learning algorithms. The tokenization process turns text into a series of symbols and words separated by white space. In other words, tokenization is the process of splitting up a sequence of letters in a text into pieces known as tokens. It helps in understanding the meaning of the text. The tokens generated can be used for various purposes, such as topic modeling, sentiment analysis, part of speech tagging, and named entity recognition. In this stage, the sentence is split into words using the tokenization technique by separating each word in the sentence.

For example, if a tweet is "I love to eat grilled chicken when I get back home tired; just grilled chicken is all that can make me happy," tokens will be: ['I', 'love', 'to', 'eat', 'grilled', 'chicken', 'when', 'I', 'get', 'back', 'home', 'tired', 'just', 'grilled', 'chicken', 'only', 'that', 'can', 'make', 'me', 'happy'].

3.2.3. Lowercase conversion

Lowercase conversion is a preprocessing step that converts all the text to lowercase. This step is essential because it helps standardize the text by removing any variations in the case of the text. Additionally, this step aids in preventing any discrepancies in the analysis due to words with different cases. That makes words in the same format to avoid changeable word formats for optimizing our processes. The lowercase conversion step is usually applied after the tokenization step.

For example, after tokenization, this is the result: ['I', 'love', 'to', 'eat', 'grilled', 'chicken', 'when', 'I', 'get', 'back', 'home', 'tired', 'just', 'grilled', 'chicken', 'only', 'that', 'can', 'make', 'me', 'happy'] , so after the lower conversion step it will be:

['i', 'love', 'to', 'eat', 'grilled', 'chicken', 'when', 'i', 'get', 'back', 'home', 'tired', 'just', 'grilled', 'chicken', 'is', 'all', 'that', 'can', 'make', 'me', 'happy'].

3.2.4. Stop word removal

Stop words are commonly used words in a language with no significant meaning in the context of the text. Examples of stop words include "the," "and," "of", "to", 'ours', 'had', 'she' 'when' etc. Stop word removal is also a crucial preprocessing step in NLP that involves removing these words from the text. The purpose of removing stop words is to reduce the text corpus's size and focus on the meaningful words in the text. The removal of stop words can improve the accuracy of the analysis and reduce the computational complexity of the algorithms used. The most common undesirable term is a punctuation mark, a pronoun, a prepositional term, or a stop word. Many pronouns and prepositions appear in English-language sentences, but these rarely contribute to machine learning. Eliminating these punctuation marks, pronouns, and prepositions enhances the performance and accuracy of the analysis and reduces the computational complexity of the algorithms used.

For example, after lowercase conversion, this is the result : ['i', 'love', 'to', 'eat', 'grilled', 'chicken', 'when', 'i', 'get', 'back', 'home', 'tired', 'just', 'grilled', 'chicken', 'is', 'all', 'that', 'can', 'make', 'me', 'happy']So after removing the stop words, it will be :
['love', 'eat', 'grilled', 'chicken', 'get', 'back', 'home', 'tired', 'grilled', 'chicken', 'make', 'happy'].

3.2.5. Stemming and lemmatization

Texts use many word forms for grammatical reasons and to identify the intended meaning. There are several word families generated from similar words, which increases the number of data pairs and reduces the effectiveness of text extraction. It is extremely useful to return such words to their root as one meaning to enhance text extraction performance; this is called the stemming and lemmatization process.

Stemming involves removing the suffixes from the words to obtain the root form, while lemmatization involves using a dictionary to identify the base form of the words. Both are used to avoid a duplicate of the same word.

Also, stemming is a common technique used for tweet data, as it helps reduce the text corpus's dimensionality by grouping words with similar meanings. That is particularly useful in sentiment analysis, where the focus is on the sentiment expressed in the text rather than the specific words used. However, lemmatization can also be useful in tweet data as it helps identify the base form of words, which can be important in some cases. For example, suppose you are interested in analyzing the topics discussed in tweets. In that case, lemmatization can help group words with similar meanings and more accurately represent the topics.

For example, after removing stop words, this is the result : ['love', 'eat', 'grilled', 'chicken', 'get', 'back', 'home', 'tired', 'grilled', 'chicken', 'make', 'happy']. So after stemming and lemmatization, it will be: ['love', 'eat', 'grill', 'chicken', 'get', 'back', 'home', 'tire', 'grill', 'chicken', 'make', 'happi'].

3.3. Sentiment analysis (SA)

Tweets are preprocessed to eliminate unnecessary information, and lexicon-based sentiment analysis using TextBlob and the Valence Aware Dictionary for Sentiment Reasoner (VADER) is performed. SA is a natural language processing technique that involves analyzing and classifying the subjective content of a text into positive, negative, or neutral sentiments. This technique is commonly used to determine the sentiment expressed in social media posts, reviews, or customer feedback.

SA has several applications in various fields, including business, politics, healthcare, and advertising. In business, SA is used to analyze customer feedback and reviews to understand the customers' sentiments towards a product or service. In politics, SA is used to analyze public opinion toward political candidates or policies. In healthcare, SA is used to analyze patients' sentiments toward their medical conditions and treatments. In advertisements for restaurants for example SA is used to fetch appropriate users already interested in the categories of food that are presented by these restaurants.

SA involves several steps, starting with data preprocessing, where the text data is cleaned, tokenized, and normalized to remove noise and irrelevant information. The next step is feature extraction, where the important features of the text data, such as words or phrases, are identified and represented in a numerical format, such as Bag of Words or TF-IDF. After feature extraction, a classification algorithm is used to classify the text into positive, negative, or neutral sentiments. The most commonly used algorithms in SA are machine learning algorithms such as Naive Bayes, Support Vector Machines, and Neural Networks.

So after completing the preprocessing step, the FIA model worked for text analysis by calculating a sentiment score from the provided document using lexicon-based approaches, with 0 or above denoting the positive sentiment, 0 or below denoting the negative sentiment, and 0 to denoting neutral.

3.3.1. lexicon-based approaches (TextBlob and VADER)

The lexicon approach has a mapping between words and sentiment, and the overall sentiment of a sentence is determined by combining the sentiment of each individual term. The lexicon-based approach provides a polarity score on a scale of -1 to 1, with -1 indicating a highly negative sentiment and 1 indicating a highly positive sentiment. Scores close to 0 indicate a neutral sentiment. Two of the popular lexicon-based approaches, which contain predefined dictionaries or rules, are TextBlob and VADER [28, 29]. It simplifies NLP tasks for textual data.

TextBlob returns polarity and subjectivity as three outputs for a given sentence. Polarity provides three sentiments: -1 for negative sentiment, +1 for positive sentiment and 0 for neutral. Subjectivity indicates subjects or judgments.

VADER is an open-source sentiment analysis method that is designed to analyze sentiments expressed in social media using lexicon and rule-based approaches. It is very intelligent for NPL tasks. It considers word order and degree modifiers in its analysis as well [30].

In this study, according the accuracy in Table 1 the FIA model used the VADER Sentiment approach to classify the sentiments expressed in Twitter data related to food.

3.4. Classification using machine learning (ML)

Classification using ML algorithms involves training algorithms on labeled data to accurately predict the class of unseen data points. Popular ML algorithms for classification include Naive Bayes, Support Vector Machines (SVM), Decision Trees, Random Forests, and Logistic Regression. These algorithms use various techniques, such as probabilities, hyperplane separation, and decision tree-based approaches. Classification with ML algorithms is widely applied in fields like image recognition, sentiment analysis, medical diagnosis, and fraud detection, facilitating automated and efficient decision-making.

3.5. Feature extraction

The preprocessed text would be subjected to feature extraction by the system, which can be accomplished through methods such as the Bag of Words model or the TF-IDF model. This stage focuses on various methods utilized for feature extraction. For the purposes of this research, TF-IDF with unigram is used to select features. Selecting the most appropriate feature among all the features can be a challenging task. The dataset may contain numerous irrelevant attributes that must be removed before further processing and analysis. Due to the abundance of features, several classification algorithms may not be able to function optimally and produce satisfactory results. Therefore, a feature selection technique must be employed to improve the overall performance of the system [31].

3.5.1. Bag of words

The Bag of Words technique is a simple and widely used approach to preprocessing text data for analysis. This technique involves counting the frequency of each word in the text corpus and representing the text as a matrix of word counts, where each row represents a document and each column represents a word in the corpus. This technique does not consider the context of the words in the text and treats each word as an independent feature. Bag of Words is often used for text classification tasks like sentiment analysis or topic modeling.

3.5.2. TF-IDF

TF-IDF is a more advanced technique than Bag of Words that considers the importance of each word in the text corpus. TF-IDF assigns a weight to each word based on its frequency in the document and its rarity in the corpus. This technique considers the context of the words in the text and gives more weight to words that are important to the meaning of the text. The TF-IDF weight is calculated as the product of the term frequency (TF) and the inverse document frequency (IDF). TF-IDF is often used for information retrieval tasks, such as sentiment analysis search engines, topic modeling, or recommendation systems.

The main difference between Bag of Words and TF-IDF is that Bag of Words treats each word as an independent feature and assigns equal weight to all words in the corpus, whereas TF-IDF assigns a weight to each word based on its frequency in the document and its rarity in the corpus. Bag of Words is simple and easy to implement, but it may not capture the importance of each word in the text. On the other hand, TF-IDF is more advanced and considers the importance of each word in the text, but it is more computationally expensive and may not be suitable for large text corpora.

To assign weights to individual words, we employ the TF-IDF approach, which aims to reflect the significance of a word within a given document collection. This approach evaluates a term's frequency of occurrence in a specific document known as the "term frequency" and adjusts it based on the prevalence of the term across all documents in the corpus known as the "inverse document frequency". Common words, such as stop-words, that appear frequently in all documents are given less weight through this process to avoid skewing the results. We utilize the Scikit-learn library to implement the TF-IDF calculation, which involves the following steps:

$$w_{ij} = tf_{ij} \times \log\left(\frac{N}{df_i}\right),$$

In the calculation of TF-IDF, the weight of a word i in a given document j is determined by the term frequency (tf ij) of word i in document j, the document frequency (df i) of word i across all documents, and the total number of documents (N) in the corpus. Specifically, df i represents the count of documents that contain the word i.

3.6. Topic modelling using Latent Dirichlet Allocation (LDA)

The objective of topic modeling is to identify clusters of related words or topics that appear frequently in tweets. In contrast to machine learning models that classify sentiment and only indicate the type of sentiment, topic modeling reveals the most commonly used words in the text. When applied to food-related tweets, it can highlight both positive and negative aspects of food companies' menus and services. Companies can then use this information to improve the quality of their services.

The literature review revealed that the accuracy of using LDA to group tweets into a common topic was low and that those matching with topics needed improvement. LDA is a versatile generative probabilistic model for unsupervised topic modeling used for textual data documents [26]. LDA can also be applicable as it can handle the unlabeled data in the dataset, extract ambiguity from English language text, and diversity for the varied writing styles of Twitter users [26].

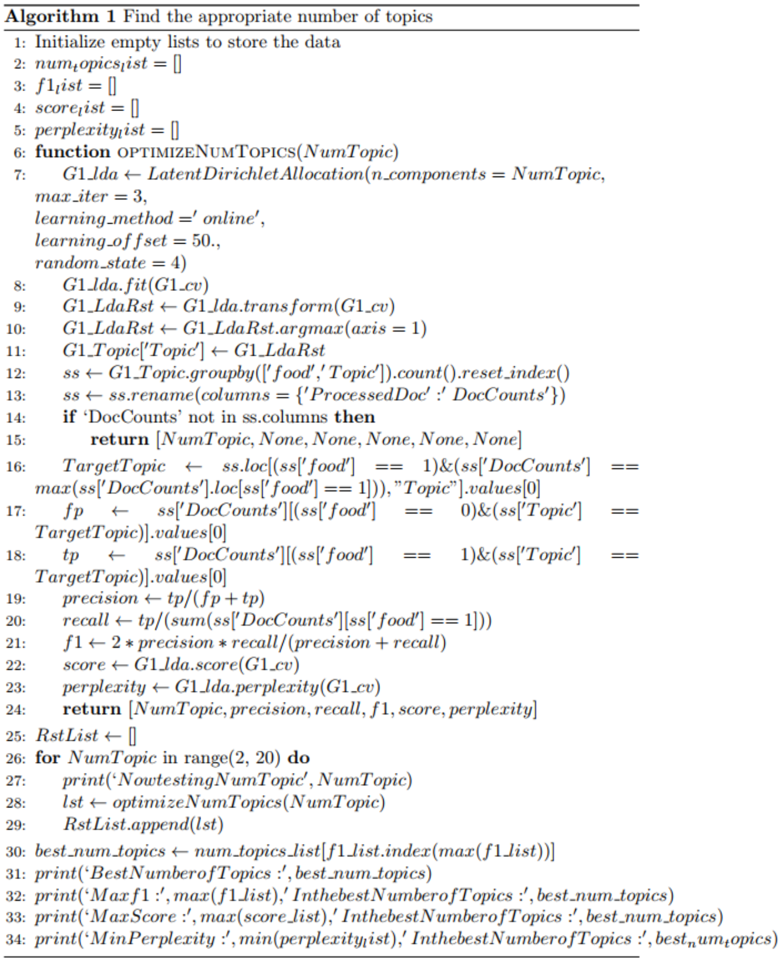
Topic modeling can be utilized to retrieve information and select features from unstructured text in the context of information retrieval. As a topic modeling algorithm, LDA is valuable in organizing extensive amounts of textual data into intersecting clusters of documents that differ from rule-based text mining techniques that rely on a dictionary or regular expression-based keyword searches [26]. LDA is a flexible probabilistic topic model capable of handling discrete data sets. It represents documents as a collection of diverse topics with varying probabilities assigned to each topic, and a list of words with corresponding probabilities defines each topic. It is important to note that a given document may contain a single topic or multiple topics with varying proportions.

As an example, consider a corpus with three documents (d1, d2, and d3) and the objective of generating three topics (t1, t2, and t3). In this scenario, d1 may be predominantly associated with t1, partially associated with t2, and minimally associated with t3. Conversely, d2 could have a mix of t1 and t2, while d3 could only be related to t3.

Despite not providing explicit direct information, LDA can still be utilized to extract valuable insights about the tweets. Furthermore, the number of topics (k) produced by LDA is typically smaller than the overall vocabulary size (V).

3.6.1. Latent Dirichlet Allocation (LDA)

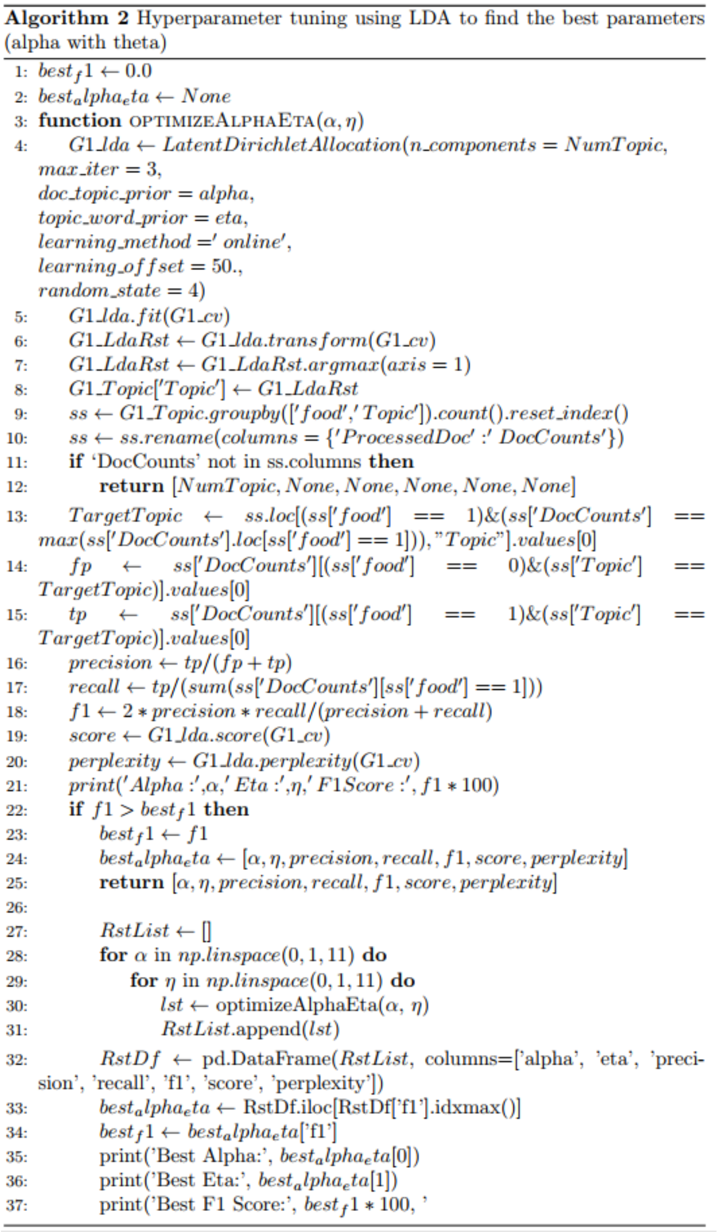
In Algorithm 1, the FIA model performs topic modeling on a dataset and tries to find the optimal number of topics. It initializes empty lists to store the results and then defines a function that takes the number of topics as input by iterating through a range of numbers from 2 to 19 and outputs several performance metrics, including precision, recall, F1-score, score, and perplexity. The function uses the Latent Dirichlet Allocation (LDA) algorithm to fit the model and then calculates the performance metrics for each topic. The function returns the performance metrics as a list. Finally, the model loops through a range of possible topics and calls the function for each one.



Algorithm 1 Find the appropriate number of topics

```
Initialize empty lists to store the data
$num_topics_list = []$
$f1_list = []$
$score_list = []$
$perplexity_list = []$
function OPTIMIZENUMTOPICS($NumTopic$)
    $G1\_lda \leftarrow LatentDirichletAllocation(n\_components = NumTopic,$
$max\_iter = 3,$
$learning\_method =' online',$
$learning\_offset = 50.,$
$random\_state = 4)$
    $G1\_lda.fit(G1\_cv)$
    $G1\_LdaRst \leftarrow G1\_lda.transform(G1\_cv)$
    $G1\_LdaRst \leftarrow G1\_LdaRst.argmax(axis = 1)$
    $G1\_Topic['Topic'] \leftarrow G1\_LdaRst$
    $ss \leftarrow G1\_Topic.groupby(['food','Topic']).count().reset\_index()$
    $ss \leftarrow ss.rename(columns = \{'ProcessedDoc' :' DocCounts'\})$
    if ‘DocCounts’ not in ss.columns then
        return $[NumTopic, None, None, None, None, None]$
    $TargetTopic \leftarrow ss.loc[(ss['food'] == 1)\&(ss['DocCounts'] == max(ss['DocCounts'].loc[ss['food'] == 1])),"Topic"].values[0]$
    $fp \leftarrow ss['DocCounts'][(ss['food'] == 0)\&(ss['Topic'] == TargetTopic)].values[0]$
    $tp \leftarrow ss['DocCounts'][(ss['food'] == 1)\&(ss['Topic'] == TargetTopic)].values[0]$
    $precision \leftarrow tp/(fp + tp)$
    $recall \leftarrow tp/(sum(ss['DocCounts'][ss['food'] == 1]))$
    $f1 \leftarrow 2 * precision * recall/(precision + recall)$
    $score \leftarrow G1\_lda.score(G1\_cv)$
    $perplexity \leftarrow G1\_lda.perplexity(G1\_cv)$
    return $[NumTopic, precision, recall, f1, score, perplexity]$
$RstList \leftarrow []$
for $NumTopic$ in range(2, 20) do
    $print(‘NowtestingNumTopic', NumTopic)$
    $lst \leftarrow optimizeNumTopics(NumTopic)$
    $RstList.append(lst)$
$best\_num\_topics \leftarrow num\_topics\_list[f1\_list.index(max(f1\_list))]$
$print(‘BestNumberofTopics :', best\_num\_topics)$
$print(‘Maxf1 :', max(f1\_list),' InthebestNumberofTopics :', best\_num\_topics)$
$print(‘MaxScore :', max(score\_list),' InthebestNumberofTopics :', best\_num\_topics)$
$print(‘MinPerplexity :', min(perplexity_list),' InthebestNumberofTopics :', best_num_topics)$
```

In Algorithm 2, the FIA Model implementation for hyperparameter tuning on a Latent Dirichlet Allocation (LDA) model, which is a type of topic modeling algorithm used in natural language processing. The optimizeAlphaEta() function is used to tune the hyperparameters alpha and eta by iteratively training and testing the LDA model with the provided training data. The training process includes several steps, such as fitting the LDA model with the training data, calculating the F1 score, precision, recall, score, and perplexity, and updating the global best_f1 and best_alpha_eta variables if the current F1 score is better than the previous best score. The nested loops iterate over different values of alpha and eta, and for each combination, the optimizeAlphaEta(alpha, eta) function is called, and the results are stored to get the best performance and accuracy for LDA.



Algorithm 2 Hyperparameter tuning using LDA to find the best parameters (alpha with theta)

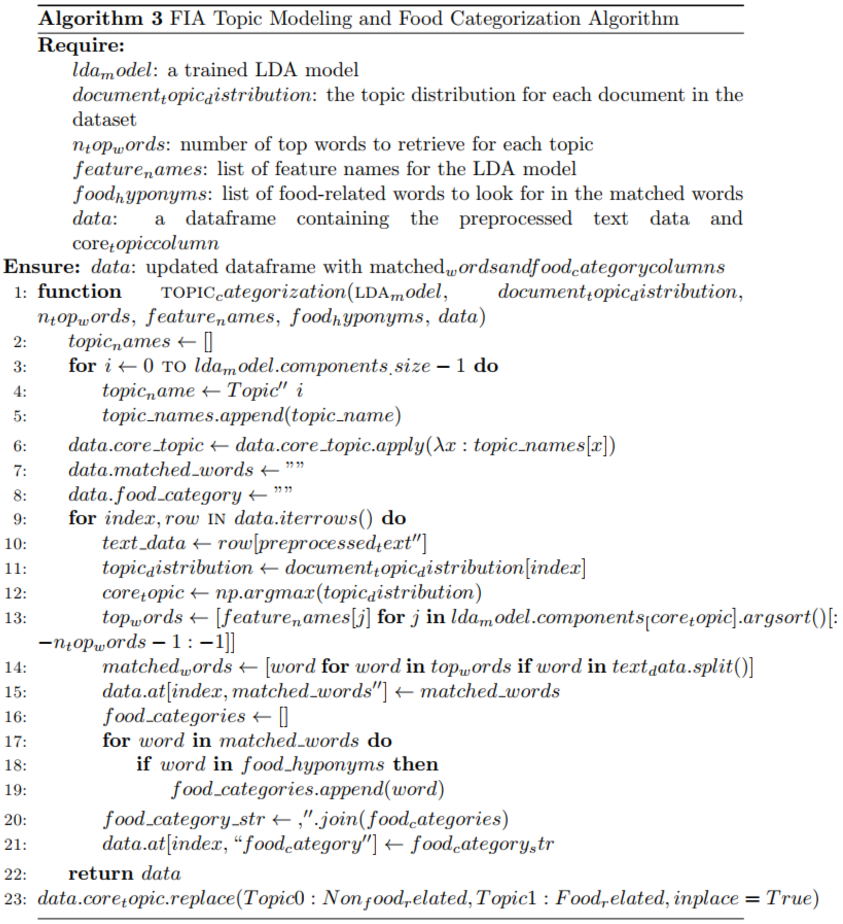
```
$best_f1 \leftarrow 0.0$
$best_alpha_eta \leftarrow None$
function OPTIMIZEALPHAETA($\alpha, \eta$)
    $G1\_lda \leftarrow LatentDirichletAllocation(n\_components = NumTopic, max\_iter = 3, doc\_topic\_prior = alpha, topic\_word\_prior = eta, learning\_method =' online', learning\_offset = 50., random\_state = 4)$
    $G1\_lda.fit(G1\_cv)$
    $G1\_LdaRst \leftarrow G1\_lda.transform(G1\_cv)$
    $G1\_LdaRst \leftarrow G1\_LdaRst.argmax(axis = 1)$
    $G1\_Topic['Topic'] \leftarrow G1\_LdaRst$
    $ss \leftarrow G1\_Topic.groupby(['food','Topic']).count().reset\_index()$
    $ss \leftarrow ss.rename(columns = \{'ProcessedDoc' :' DocCounts'\})$
    if ‘DocCounts’ not in ss.columns then
        return [$NumTopic, None, None, None, None, None$]
    $TargetTopic \leftarrow ss.loc[(ss['food'] == 1)\&(ss['DocCounts'] == max(ss['DocCounts'].loc[ss['food'] == 1])), "Topic"].values[0]$
    $fp \leftarrow ss['DocCounts'][(ss['food'] == 0)\&(ss['Topic'] == TargetTopic)].values[0]$
    $tp \leftarrow ss['DocCounts'][(ss['food'] == 1)\&(ss['Topic'] == TargetTopic)].values[0]$
    $precision \leftarrow tp/(fp + tp)$
    $recall \leftarrow tp/(sum(ss['DocCounts'][ss['food'] == 1]))$
    $f1 \leftarrow 2 * precision * recall/(precision + recall)$
    $score \leftarrow G1\_lda.score(G1\_cv)$
    $perplexity \leftarrow G1\_lda.perplexity(G1\_cv)$
    $print('Alpha :',\alpha,' Eta :',\eta,' F1Score :', f1 * 100)$
    if $f1 > best_f1$ then
        $best_f1 \leftarrow f1$
        $best_alpha_eta \leftarrow [\alpha, \eta, precision, recall, f1, score, perplexity]$
        return [$\alpha, \eta, precision, recall, f1, score, perplexity$]

        $RstList \leftarrow []$
        for $\alpha$ in $np.linspace(0, 1, 11)$ do
            for $\eta$ in $np.linspace(0, 1, 11)$ do
                $lst \leftarrow$ optimizeAlphaEta($\alpha$, $\eta$)
                $RstList$.append($lst$)
        $RstDf \leftarrow$ pd.DataFrame($RstList$, columns=[’alpha’, ’eta’, ’precision’, ’recall’, ’f1’, ’score’, ’perplexity’])
        $best_alpha_eta \leftarrow$ RstDf.iloc[RstDf[’f1’].idxmax()]
        $best_f1 \leftarrow best_alpha_eta$[’f1’]
        print(’Best Alpha:’, $best_alpha_eta$[0])
        print(’Best Eta:’, $best_alpha_eta$[1])
        print(’Best F1 Score:’, $best_f1 * 100$, ’
```

3.6.2. FIA Topic Modeling and Food Categorization

Algorithm 3 extracts food-related tweets from the dataset using topic modeling and keyword matching techniques in some steps: -

- Identify the core topics of each tweet using LDA and replace them with corresponding topic names.
- Create two new columns: one for matched words and another for food categories.
- Loop through each tweet in the dataset and:
    - Identify the core topic.
    - Retrieve the top words associated with that topic.
    - Match the top words with the words in the tweet.
    - Update the matched words column with the list of matched words.
    - Identify food categories for each tweet by searching for food-related hyponyms in the matched word list.
    - Combine food categories into a comma-separated string and add it to the food category column.
- Update core topic names for better readability.
- Print the topic distribution, sentiment score, sentiment category, and matching words for each food-related tweet.



Algorithm 3 FIA Topic Modeling and Food Categorization Algorithm

```
Require:
    $lda_{m}odel$: a trained LDA model
    $document_{t}opic_{d}istribution$: the topic distribution for each document in the dataset
    $n_{t}op_{w}ords$: number of top words to retrieve for each topic
    $feature_{n}ames$: list of feature names for the LDA model
    $food_{h}yponyms$: list of food-related words to look for in the matched words
    $data$: a dataframe containing the preprocessed text data and core$_{t}opiccolumn$
Ensure: $data$: updated dataframe with matched$_{w}ordsandfood_{c}ategorycolumns$
function TOPIC$_{c}ategorization$(LDA$_{m}odel$, $document_{t}opic_{d}istribution$, $n_{t}op_{w}ords$, $feature_{n}ames$, $food_{h}yponyms$, $data$)
    $topic_{n}ames \leftarrow []$
    for $i \leftarrow 0$ TO $lda_{m}odel.components\_size - 1$ do
        $topic_{n}ame \leftarrow Topic'' \ i$
        $topic\_names.append(topic\_name)$
    $data.core\_topic \leftarrow data.core\_topic.apply(\lambda x : topic\_names[x])$
    $data.matched\_words \leftarrow ""$
    $data.food\_category \leftarrow ""$
    for $index, row$ IN $data.iterrows()$ do
        $text\_data \leftarrow row[preprocessed_{t}ext'']$
        $topic_{d}istribution \leftarrow document_{t}opic_{d}istribution[index]$
        $core_{t}opic \leftarrow np.argmax(topic_{d}istribution)$
        $top_{w}ords \leftarrow [feature_{n}ames[j]$ for $j$ in $lda_{m}odel.components_{[}core_{t}opic].argsort()[: -n_{t}op_{w}ords - 1 : -1]]$
        $matched_{w}ords \leftarrow [word$ for $word$ in $top_{w}ords$ if $word$ in $text_{d}ata.split()]$
        $data.at[index, matched\_words''] \leftarrow matched\_words$
        $food\_categories \leftarrow []$
        for $word$ in $matched\_words$ do
            if $word$ in $food\_hyponyms$ then
                $food\_categories.append(word)$
        $food\_category\_str \leftarrow ,''.join(food_{c}ategories)$
        $data.at[index, “food_{c}ategory''] \leftarrow food_{c}ategory_{s}tr$
    return $data$
$data.core_{t}opic.replace(Topic0 : Non_{f}ood_{r}elated, Topic1 : Food_{r}elated, inplace = True)$
```

In 2015, [32] various evaluation metrics are utilized to evaluate the performance of the models. The efficiency of the models is determined by employing the test data, and evaluation metrics such as accuracy, precision, recall, and F1 score, are used in this study. Accuracy is determined by dividing true positive (TP) plus true negative (TN) predictions over total predictions. Precision is calculated by dividing TP predictions by TP plus false positive (FP) predictions. Similarly, recall is determined by dividing the number of TP predictions by the sum of TP and false negative (FN) predictions. Likewise, the F1 score is the harmonic mean of precision and recall and is calculated as two times the product of precision and recall divided by their sum.

Accuracy = (TP + TN) / (TP+TN+PF+FN)

Precision = TP / (TP + FP)

Recall = TP / (TP + FN)

$\text{F1 Score} = 2 * (\text{Precision} * \text{Recall})/(\text{Precision} + \text{Recall})$

4. Results and discussion

This section presents and discusses the experimental results regarding the sentiments of Twitter users towards food to extract their food preferences. The results of the lexicon-based approach for sentiment analysis and the LDA approach on the discussed topics are also included.

4.1. Sentiment analysis (SA)

As shown in Fig. 6 and 7, sentiment analysis was calculated for each tweet; a score was calculated to categorize each word as positive or negative, and the value of the score shows how positive or negative the tweet is or whether it is neutral. The performance of lexicon-based approaches, such as TextBlob and the valency-aware dictionary for sentiment reasoning (VADER), is evaluated.

In Table 1 and Fig. 2, a comparison shows the performance of five different classification algorithms for the VADER lexicon approach: Naive Bayes, Support Vector Machines, Decision Tree, Random Forest, and Logistic Regression. The proposed model was evaluated based on four metrics: accuracy, precision, recall, and F1 score. The highest F1 score in Table 1 is achieved by Logistic Regression with a value of 75.0%, closely followed by Support Vector Machines with a score of 74.4%. Naive Bayes has the lowest F1 score of 63.6%.The Decision Tree and Random Forest algorithms both achieved F1-scores in the 71% range, with 70.6% and 72.4%, respectively. Overall, the F1-scores indicate that the Logistic Regression algorithm performed the best out of the five algorithms evaluated, with an F1-score of 75.0%.

Table 1. Performance metrics comparison table for the VADER lexicon-based approach using Naive Bayes, Support Vector Machines, Decision Tree, Random Forest, and Logistic Regression for sentiment analysis.

Performance Measure Metric	Accuracy	Precision	Recall	F1 Score
Naive Bayes	65.3%	67.3%	65.3%	63.6%
Support Vector Machines	74.8%	74.6%	74.8%	74.4%
Decision Tree	70.8%	71.1%	70.8%	70.6%
Random Forest	73.2%	75.5%	73.2%	72.4%
Logistic Regression	75.5%	75.2%	75.5%	75.0%

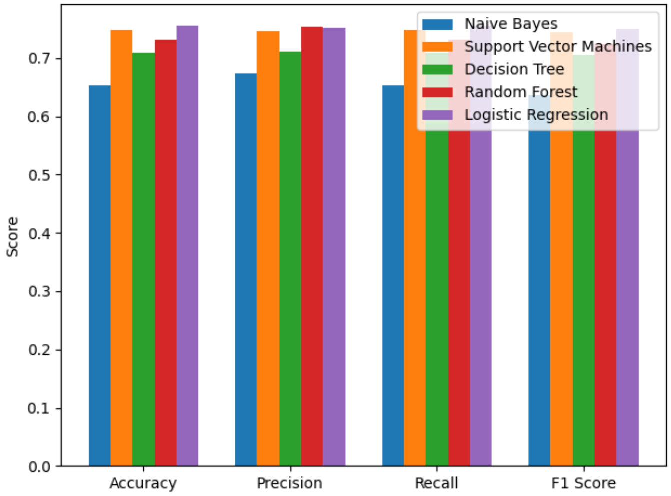


Fig. 2. Performance comparison chart for the VADER lexicon-based approach using Naive Bayes, Support Vector Machines, Decision Tree, Random Forest, and Logistic Regression for sentiment analysis.

In Table 2 and Fig. 3, a comparison shows the performance of five different classification algorithms for the TextBlob lexicon approach: Naive Bayes, Support Vector Machines, Decision Tree, Random Forest, and Logistic Regression. The proposed model was evaluated based four metrics: accuracy, precision, recall, and F1 score. The highest F1 score is achieved by the Decision Tree model with a score of 73.1%, followed by the Support Vector Machines model with a score of 71.4%. The lowest F1 score is achieved by the Naive Bayes model, with a score of 61.9%. Overall, the F1 scores of the algorithms range from 61.9% to 73.1% and indicate that the Decision Tree algorithm performed the best out of the five algorithms evaluated, with an F1 score of 73.1%.

Table 2. Performance metrics comparison table for the TextBlob lexicon-based approach using Naive Bayes, Support Vector Machines, Decision Tree, Random Forest, and Logistic Regression for sentiment analysis.

Performance Measure Metric	Accuracy	Precision	Recall	F1 Score
Naive Bayes	65.5%	65.2%	65.5%	61.9%
Support Vector Machines	71.8%	71.5%	71.8%	71.4%
Decision Tree	72.7%	73.9%	72.7%	73.1%
Random Forest	69.0%	70.1%	69.0%	68.4%
Logistic Regression	70.8%	70.6%	70.8%	70.0%

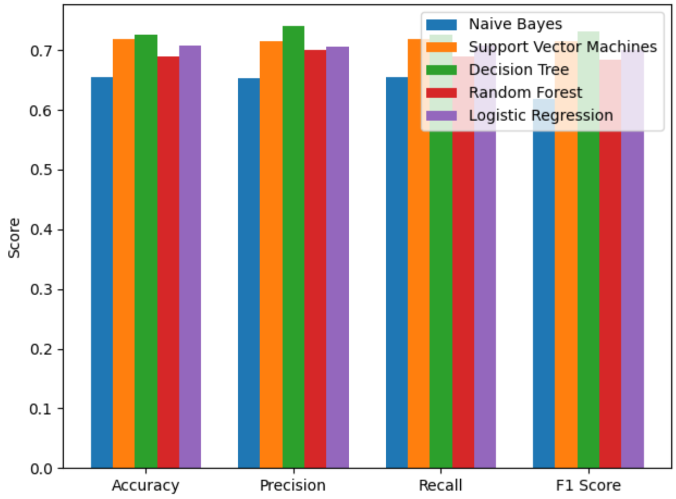


Fig. 3. Performance comparison chart for the TextBlob lexicon-based approach using Naive Bayes, Support Vector Machines, Decision Tree, Random Forest, and Logistic Regression for sentiment analysis.

According to the proposed model FIA in this study, both lexicons have an acceptable level of accuracy: 75% for VADER when using Logistic Regression and 73.1% for TextBlob when using Decision Tree. VADER emerged as a better lexicon in comparison to TextBlob based on the total performance score.

4.2. TF-IDF

TF-IDF is a method for determining the importance of a word in a document. Term The frequency of a term (t) is calculated by dividing the number of occurrences of the term in a document by the total number of words in the document. In this stage, calculate the weight of each word in the tweet to discover the core topic for each tweet applied. This is an example of the results shown in Table 3 after applying the final stage in preprocessing stemming and lemmatization ['love', 'eat', 'grill', 'chicken', 'get', 'back', 'home', 'tire', 'grill', 'chicken', 'make', 'happi']. Table 3 shows how grill and chicken have the top percentage of 0.417 which may indicate the core topic for this tweet.

Table 3. The TF-IDF results show the weight of each word in an example tweet to discover the core topic for each tweet.

Word	TF-IDF
back	0.208514414
chicken	0.417028828
eat	0.208514414
get	0.208514414
grill	0.417028828
happi	0.208514414
home	0.208514414
love	0.208514414
make	0.208514414
tire	0.208514414

4.3. Topic modelling using Latent Dirichlet Allocation (LDA)

Topic modeling means knowing which tweets to include in which topic category according to the top words for each topic. For example, a tweet might say, "I love to eat grilled chicken when I get back home tired; just grilled chicken is all that can make me happy." So it will be on a topic that has keywords about food (food-related).

In this stage, the FIA model makes preliminary tests for a topic modeling task. It initializes the number of topics to 5 and applies the LDA algorithm to fit the model using an online learning method with a maximum of 5 iterations. The resulting matrix is transformed to assign each document to its most likely topic. In Table 4, the top 10 words of each topic are then shown, and it uses the pyLDAvis library to visualize the LDA model using the t-SNE multidimensional scaling shown in Fig. 4. The resulting interactive visualization panel can be used for further analysis of the LDA model.

Table 4. Top words in The LDA topic modeling results consist of 5 number of topics initially to test results.

Topic	Keywords
0	lose, good, vote, win, god, way, flavor, time, job, elect
1	chicken, delici, eat, pasta, tasti, recip, sauc, fri, burger, tri
2	win, wont, game, goal, play, match, score, team, live, penalti
3	fan, lose, wont, like, dont, know, peopl, think, want, disgust
4	chicken, delici, amp, day, today, like, dinner, cook, celebr, fri

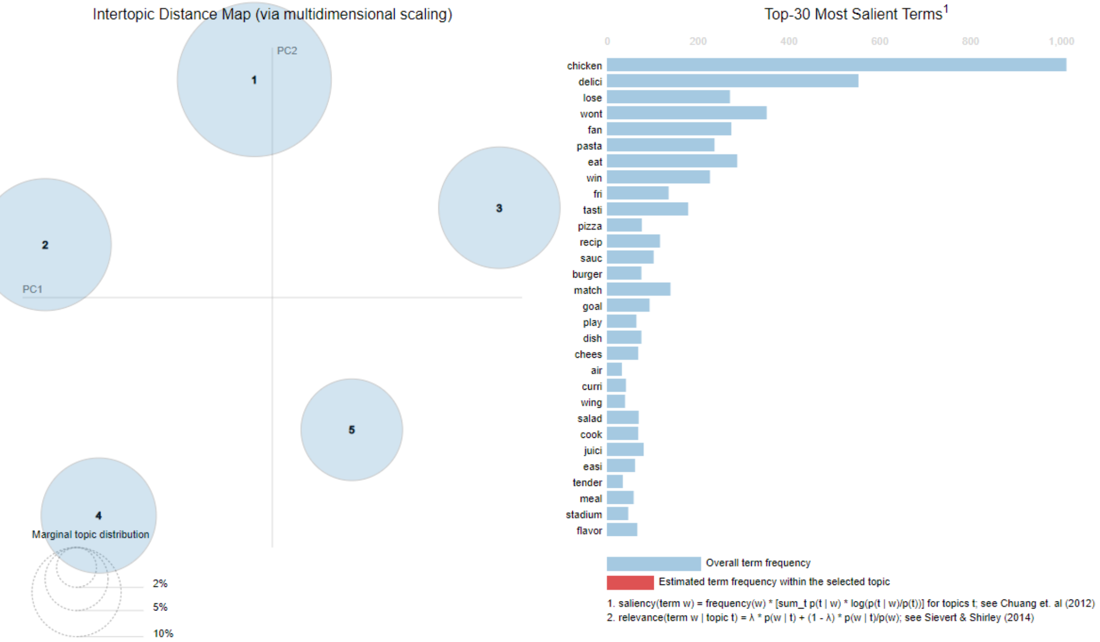


Fig. 4. The top words in the LDA topic modeling results consist of 5 topics initially tested.

As mentioned in Algorithm 1, the results are stored in the lists, and the best number of topics can be determined based on the F1 score shown in Fig. 5. A higher F1 and likelihood imply a better model; however, a lower perplexity implies a better model, so the results show the optimal number of topics is 2, as shown in Table 5 and Fig. 6. Also in Table 5, the top 25 words of each topic for related food and non-related food are shown to obtain more clear results.

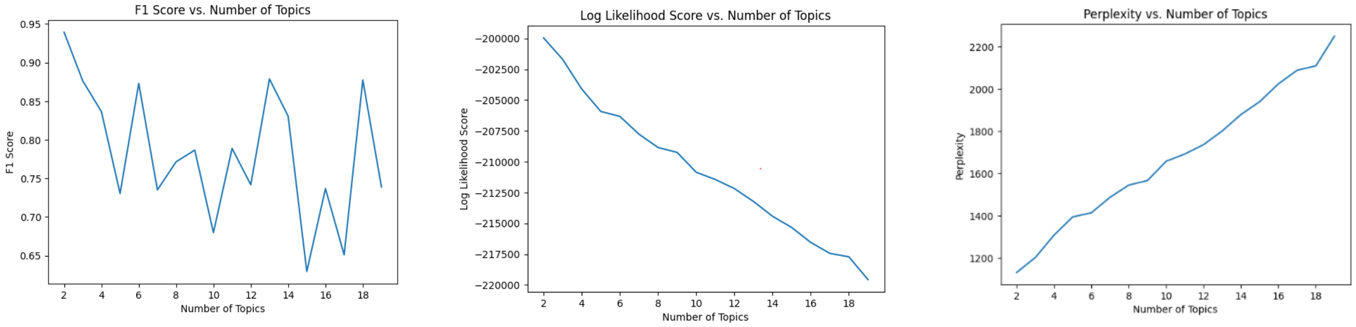


Fig. 5. Comparison of three scores measured by F1, log-likelihood, and perplexity to extract the best number of topics appropriate to the FIA proposed model.

Table 5. The top words in the LDA topic modeling results consist of 5 topics initially tested.

Topic	Keywords
0	wont, lose, fan, win, match, dont, like, know, goal, peopl, time, think, game, play, let, want, year, good, say, team, way, stadium, right, come, day
1	chicken, delici, eat, pasta, tasti, like, fri, look, recip, sauc, tri, disgust, amp, good, burger, new, today, juici, dont, love, dish, food, pizza, sandwich, day

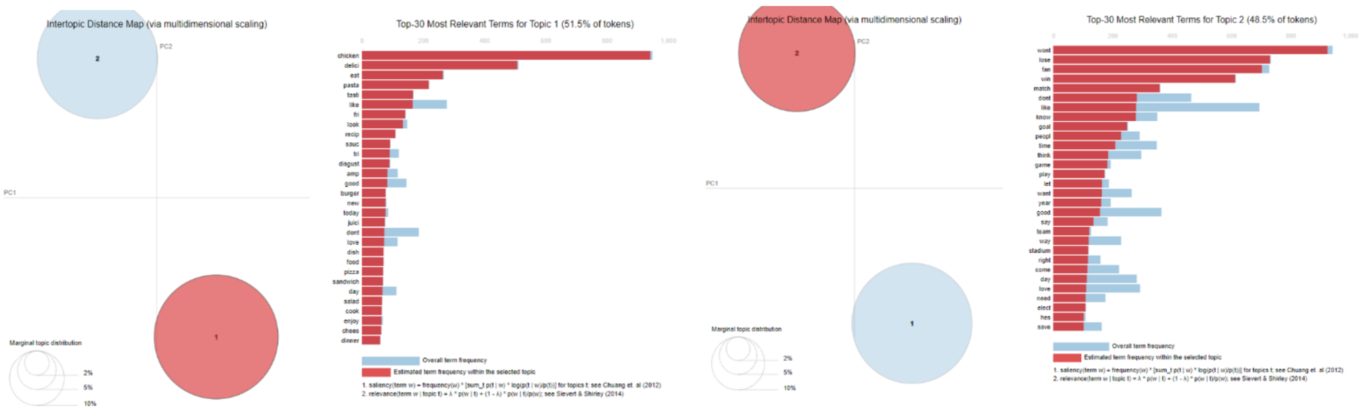


Fig. 6. The top words in the LDA topic modeling results, with the optimal number of topics being 2, were extracted from the best result of the F1 score.

As mentioned in Algorithm 2, the FIA Model methodology is implemented with hyperparameter tuning on LDA for topic modeling for optimal performance. The optimizeAlphaEta function was used to train and test the LDA model iteratively, updating the best_f1 and best_alpha_eta variables based on the F1 score and other metrics, resulting in enhanced accuracy, which indicated that the F1 score is 95% accurate when the value for alpha is 0 and the value of eta is 1, as shown in Fig. 7.

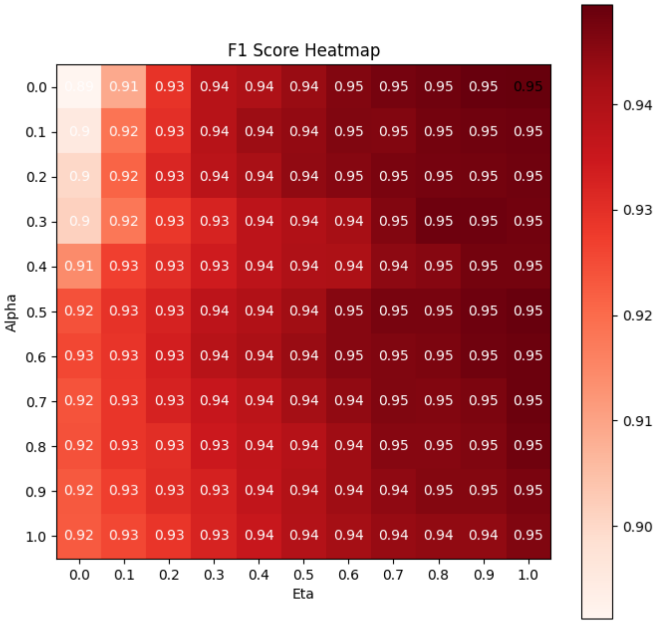


Fig. 7. Hyperparameter tuning using LDA topic modeling performance accuracy for specific Alpha and Eta, with the optimal number of topics being 2, shows that the accuracy of the F1 score is 95%.

4.4. FIA model for tweets restaurant advertisements recommendation systems

A restaurant advertisement recommendation system based on tweets is a machine learning system that aims to provide personalized restaurant recommendations based on users' food preferences and tweet history. It uses natural language processing techniques to preprocess tweets and extract features from them. The system then models user preferences using clustering or topic modeling algorithms and recommends restaurants that match the user's food preferences using recommendation techniques.

FIA is the model that is useful for any restaurant advertisement recommendation system that needs to fetch and access users' food preferences from their tweets and data about users such as user name, user id, location, etc. Therefore, it has the potential to be a useful model for both users and restaurants, helpful for providing personalized food recommendations through restaurant advertisement recommendation systems that fit users' needs and increasing customer satisfaction and engagement with customers. In addition, by using the FIA model, any restaurant can determine where it needs to establish a new restaurant location or branch. For example, if there is a restaurant for fried chicken, it needs to open a new branch in a specific location, such as California, United States. So this restaurant is available to know the number of users and their data in California, United States, that are already interested in fried chicken, as well as additional valuable data if required. Finally, FIA is diligent in ensuring that users interested in particular foods are matched with the appropriate restaurant and vice versa, which is useful to ensure the advertisement on Twitter reaches only the appropriate users to save time and cost.

The sample results of LDA and sentiment analysis on tweets posted by different users from different locations using the FIA model are shown in Table 6. Each tweet is assigned a sentiment score and sentiment label, which can be positive or negative. The tweets are related to food that is discovered on LDA and is categorized under a food category such as burger, pizza, pasta, or chicken. The table sample reveals that most users have posted tweets with a positive sentiment score, indicating that they enjoyed the food mentioned in the tweets, particularly burgers, pizza, chicken, and pasta. However, one user has posted a tweet with a negative sentiment score, suggesting a bad experience with burgers. The table also shows that the users are from different locations.

Overall, the FIA model can be useful for analyzing customer sentiment towards different food categories across multiple locations or the same food category in the same location, allowing restaurants to send relevant advertisements to users who are interested in their food category, while ignoring those who are not, thereby maximizing profit and minimizing expenses.

Table 7. Sample results for topic modeling and sentiment analysis using the FIA model to extract food-related tweets and food preference categories from different locations for users to extract food preferences for each tweet.

User Name	Location	Tweets	Sentiment Score	Sentiment	Topic	Matched words	Food category
Mokafu0225	Aberdeen United Kingdom	Tweet 1	0.6182	positive	Food-related	delici, tasti, like, burger	burger
RenFloralGhost	The Primordial Garden,India	Tweet 2	-0.6996	negative	Food-related	sauc, tasti, disgust, burger	burger
giangiskitchen	Bristol, Pennsylvania USA	Tweet 3	0.9337	positive	Food-related	delici, pasta, like, pizza	pizza, pasta
DonnaDundasBlog	Sheffield, England	Tweet 4	0.8981	positive	Food-related	chicken, delici, pasta, tasti, burger	chicken, burger, pasta
ClawDaddi	Bristol, Pennsylvania, USA	Tweet 5	0.9097	positive	Food-related	chicken, tasti, like, fri, burger	chicken, burger

5. Conclusion

Nowadays, Twitter posts are an important source of data for identifying the positive interests of users and creating intelligent recommendation systems. These posts contain a huge amount of information that can be analyzed to determine users' preferences on a variety of topics, such as food. The analysis of Twitter posts is an interesting field of study. Multiple studies have analyzed the sentiment of tweets. This research paper studied the challenges of extracting food preferences from Twitter users' posts and their locations to help restaurant owners decide where they can open new branches close to the users who are interested in their food categories. On the other hand, matching suitable restaurant advertisements that fit the user's food preferences. This research paper demonstrates the potential of analyzing Twitter posts for food interest analysis. By collecting and clustering 20,000 publicly available tweets and using a combination of the Latent Dirichlet Allocation topic modeling method and the sentiment analysis approach. LDA topic modeling method would help to find the words belonging to a topic that appeared in the Twitter posts. This paper also investigates the level of accuracy of two lexicon-based approaches to sentiment analysis for tweets. Both lexicons have acceptable levels of accuracy, with 75% for VADER using the Logistic Regression algorithm and 73.1% for TextBlob using the Decision Tree algorithm. The proposed model identifies and extracts food preference categories in which users are interested. In this regard, discovering a user's food preferences from Twitter posts could be useful for various applications that try to find restaurants that fit the user's taste. Additionally, applications that use artificial intelligence for restaurant advertisements can find suitable users who are already interested in the food categories offered by these restaurants. This presents a big opportunity for these restaurants to open branches in specific locations for users who are interested in them. Also, the data extracted from this proposed model may serve as input for Internet of Things applications to fetch users' food preferences. In future work, a recommender system will be built to suggest suitable restaurants to users that match their preferences. In addition, the recommender system recommends locations for opening new restaurant branches based on the users' preferred food categories for this restaurant.

Acknowledgment

The authors would like to thank Fayoum University for supporting the publication of this work.

Author Contributions

All authors contributed to this work. A. M. Mohamed collected the dataset of tweets and completed the experimental results and evaluations. A. M. Mohamed completed the paper writing. Both S. A. Taie and H. Al-Feel followed the paper writing, analyzing the data, validation, and performance of the results. S. A. Taie followed the revision and submission of the manuscript for publication.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] R.Gross, A.Acquisti, Information revelation and privacy in online social networks, In Proceedings of the 2005 ACM workshop on Privacy in the electronic society, (2005) 71-80.

[2] V.Kalra, R.Aggarwal, Importance of Text Data Preprocessing & in RapidMiner, Proceedings of the First International Conference on Information Technology and Knowledge Management, 14(2017) 71–75.

[3] D.Rao, D.Yarowsky, A.Shreevats, M. Gupta, Classifying latent user attributes in twitter, In Proceedings of the 2nd international workshop on Search and mining user-generated contents, (2010) 37-44.

[4] E.Cambria, S. Poria, A.Gelbukh, M.Thelwall, Sentiment Analysis Is a Big Suitcase, IEEE Intelligent Systems, 32(2017), 74-80.

[5] H.Vahdat-Nejad, S. O. Eilaki, S.Izadpanah, Towards a better understanding of ubiquitous cloud computing, International Journal of Cloud Applications and Computing (IJCAC) 8(2018), 1-20.

[6] H.Vahdat-Nejad, E. Asani, Z. Mahmoodian, M. H. Mohseni, Context-aware computing for mobile crowd sensing: A survey, Future Generation Computer Systems, 99(2019) 321-332.

[7] S.K.Trivedi, A. Singh, Twitter Sentiment Analysis of App Based Online Food Delivery Companies, Global Knowledge, Memory and Communication 70(2021) 891–910.

[8] H.M.Ahmed, M. J. Awan, N. S. Khan, A. Yasin, H. M. F. Shehzad, Sentiment Analysis of Online Food Reviews using Big Data Analytics, Elementary Education Online, 20(2021) 827-836.

[9] S.Ao, Sentiment analysis based on financial tweets and market information, International Conference on Audio, Language and Image Processing (ICALIP), IEEE, (2018) 321-326.

[10] B.P.Pokharel, Twitter Sentiment Analysis During Covid-19 Outbreak in Nepal, Social Science Research Network,(2020).

[11] W.Wahyuni, Implementation of the Support Vector Machine Method for Sentiment Analysis Using Twitter Data, Knowbase: International Journal of Knowledge in Database 2(2022) 166-180.

[12] A.R. Prananda, I. Thalib, Sentiment Analysis for Customer Review: Case Study of GO-JEK Expansion, Journal of Information Systems Engineering and Business Intelligence, 6(2020) 1.

[13] A. Khattak, R. Batool, F. A. Satti, J. Hussain, W. A. Khan, A. Khan, B. Hayat, Tweets Classification and Sentiment Analysis for Personalized Tweets Recommendation, Complexity, (2020) 1–11.

[14] M.A.Hadi, N. Anwar, B. Tjahjono, B. A. Sekti, Y. F. Achmad, Go-Food Sentiment Analysis Using Twitter Data, Compared the Performance of the Random Forest Algorithm with That of the Linear Support Vector Classifier, First Mandalika International Multi-Conference on Science and Engineering 2022, MIMSE (Informatics and Computer Science)(MIMSE-IC-2022), (2022) 3-13.

[15]D. Maier, A. Waldherr, P. Miltner, G. Wiedemann, A. Niekler, A. Keinert, B. Pfetsch, Applying LDA Topic Modeling in Communication Research: Toward a Valid and Reliable Methodology, Communication Methods and Measures, 12 (2018) 93–118.

[16] D. Naskar, S. Mokaddem, M. Rebollo, E. Onaindia, Sentiment analysis in social networks through topic modeling, In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), (2016) 46-53

[17] H. Jelodar, Y.Wang, C. Yuan, X. Feng, X. Jiang, Y. Li, L. Zhao, Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey, Multimedia Tools and Applications, 78(2019)15169–15211

[18] F. Zarrinkalam, H. Fani, E. Bagheri, M. Kahani, W. Du, Semantics-enabled user interest detection from Twitter, In 2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), 1(2015) 469-476.

[19] N. Mangal, R. Niyogi, A. Milani, Analysis of users' interest based on tweets, In Computational Science and Its Applications–ICCSA 2016: 16th International Conference, Beijing, China, July 4-7, 2016, Proceedings, 16(2016)12-23.

[20] V.Vallurupalli, I.Bose, Exploring Thematic Composition of Online Reviews: A Topic Modeling Approach, Electronic Markets 30 (2020) 791–804.

[21] F. ul Mustafa, I. Ashraf, A. Baqir, U. Ahmad, S. Malik and S. Mehmood.. 2020."Prediction of user's interest based on urdu tweets." In 2020 International Symposium on Recent Advances in Electrical Engineering & Computer Sciences (RAEE & CS), vol. 5, pp. 1-6. IEEE. https://doi.org /10.1109/RAEECS50817.2020.9265694.

[22] A. Milani, N. Rajdeep, N. Mangal, R. K. Mudgal, V. Franzoni, Sentiment Extraction and Classification for the Analysis of Users Interest in Tweets, International Journal of Web Information Systems, (2018)

[23]E. Asani, H. Vahdatnejad, S. Hosseinabadi, J. Sadri, Extracting user's food preferences by sentiment analysis, 8th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS), IEEE, (2020) 066-069.

[24] C. Gokulnath, M. K. Priyan, E. V. Balan, KP. Rama Prabha, R. Jeyanthi, Preservation of privacy in data mining by using PCA based perturbation technique, International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), IEEE,(2015) 202-206.

[25] M. K. A.Venkataramaiah, N. A. N. Achar, Twitter sentiment analysis using aspect-based bidirectional gated recurrent unit with self-attention mechanism, International Journal of Intelligent Engineering and Systems 13 (2020) 97–110.

[26] W.Anwar, I. S. Bajwa, M. A. Choudhary, S. Ramzan, An Empirical Study on Forensic Analysis of Urdu Text Using LDA-Based Authorship Attribution, IEEE Access 7(2019) 3224–34.

[27]A. El Kah, I. Zeroual, The Effects of Pre-Processing Techniques on Arabic Text Classification, International Journal of Advanced Trends in Computer Science and Engineering 10 (2021) 41–48.

[28]G. Chandrasekaran, D. J. Hemanth, Deep Learning and TextBlob Based Sentiment Analysis for Coronavirus (COVID-19) Using Twitter Data, International Journal on Artificial Intelligence Tools, 31 (2022).

[29] R.D. Endsuy, Sentiment Analysis between VADER and EDA for the US Presidential Election 2020 on Twitter Datasets, Journal of Applied Data Sciences 2 (2021) 08–18.

[30] V.S.Chauhan, , A. Bansal, A. Goel, Twitter Sentiment Analysis Using Vader, International Journal of Advance Research, Ideas and Innovations in Technology 4 (2018) 485–89.

[31] M. Ali, A. Baqir, G. Psaila, S. Malik,Towards the Discovery of Influencers to Follow in Micro-Blogs (Twitter) by Detecting Topics in Posted Messages (Tweets), Applied Sciences 10, no. 16(2020) 5715.

[32] M. Hossin, M. N. Sulaiman, A Review on Evaluation Metrics for Data Classification Evaluations, International Journal of Data Mining & Knowledge Management Process 5 (2015) 01–11.