A Recommendation Model Based on Visitor Preferences on Commercial Websites Using the TKD-NM Algorithm

Piyanuch Chaipornkaew(✉) and Thepparit Banditwattanawong

Department of Computer Science, Kasetsart University, Bangkok, Thailand
{piyanuch.chai,thepparit.b}@ku.th

Abstract. In recent years, recommendation models have been widely implemented in various areas. In order to construct recommendation models, much research makes use of machine learning techniques. This research also proposes a recommendation model using a novel machine learning technique called the "TKD-NM" algorithm. It is the combination of the TF-IDF, KMeans, and Decision Tree incorporated with the Nelder-Mead algorithm. TF-IDF was applied to form word vectorization from webpage headings. KMeans was utilized for clustering webpage headings while the Decision Tree algorithm was employed to investigate the performance of KMeans. Nelder-Mead was applied to find the optimum values of word vectorization. The dataset analyzed in the research was collected from a specific commercial website. Visitor preferences on the website were considered as the dataset in the research. The recommendation lists were retrieved from webpages in the same cluster. The prediction accuracy of the TKD-NM algorithm was approximately 97.31% while the prediction accuracy of the baseline model was only 88.87%.

Keywords: Recommendation model · Webpage heading · TKD-NM · TF-IDF · KMeans · Decision Tree · Nelder-Mead

1 Introduction

Steve Jobs said "You can't ask customers what they want and then try to give that to them. By the time you get it built, they'll want something new." [1]. In order to provide something new that meets customer needs, data related to customer preferences should be considered and analyzed. In the case of commercial websites, customer preferences could be extracted from the content of their visited webpages. Once customer needs are known, it is possible to suggest new contents which relate to customer interests. Such information can be offered via a website which is hereafter referred to as "an intelligent website". The definition of an intelligent website is a website that can offer what customers need at an early stage without any request from them. In order to implement an intelligent website, machine learning techniques are required.

In the case of commercial websites, machine learning techniques are applied in many ways. Data obtained from back-end websites are analyzed using machine learning and

I. Awan et al. (Eds.): Deep-BDB 2021, LNNS 309, pp. 123–133, 2022.
https://doi.org/10.1007/978-3-030-84337-3_10

turned into useful information. Data visualization of results from machine learning could help executive-level managers in decision-making. Machine learning is also applied to construct a recommendation model which can assist in the planning of business campaigns. The definition of a recommendation system is a system that is used to find new items or services that are related to the interests of users [2]. Such systems could help administrators to drive businesses more effectively. The recommendation system is applied with several goals, such as maximizing profits, minimizing cost, and minimizing risk.

There are many widely-used machine learning techniques to implement a recommendation system. Examples of machine learning algorithms are KMeans, LDA, deep neural network, Doc2Vec, ensemble, matrix factorization, decision tree, SVM, and TF-IDF [3–5]. The top five application domains of the recommendation system are movie, social, academic, news, and e-commerce domains [6]. This research falls under the last domain, which is a recommendation system for e-commerce.

This paper proposes a recommendation model based on visitor preferences on commercial websites using a novel algorithm, the "TKD-NM Algorithm". This algorithm is a combination of TF-IDF, K-Means, and Decision Tree incorporated with the Nelder-Mead algorithm. The proposed model was divided into three phases: data preprocessing, clustering optimization, and webpage suggestion. The result from the proposed model is a recommendation list of webpages in the same cluster.

The remainder of the paper is as follows. Section 2 is the literature review. Section 3 presents the methodology. Section 4 presents the experimental results. Section 5 provides the conclusion and suggestions, and last section presents the acknowledgements.

2 Literature Review

Recommendation models are generated from various machine learning techniques, such as clustering, classification, and association. Erra et al. conducted research on topic modelling using the TF-IDF algorithm. The paper proposed an approximate TF-IDF to extract topics from a massive message stream using GPU [7]. There were two main contributions of the research. Firstly, an approximate TF-IDF measure was implemented. Secondly, parallel implementations of the calculation of the approximate TF-IDF based on GPUs were introduced. The research concluded that the parallel GPU architecture met the fast response requirements and overcame storage constraints when processing a continuous flow of data stream in real time. The experiment also revealed that the GPU implementation was stable and performed well even with limited memory. Furthermore, the time to compute the approximate TF-IDF measure on the GPU was not varied depending on the data source.

Another research on clustering was entitled "An evaluation of document clustering and topic modelling in two online social networks: Twitter and Reddit" [8]. The two main contributions were document clustering and topic modelling for social media text data. The research presented document and word embedding representations of online social network data. The combination of doc2vec and TF-IDF weighted meant word embedding representations delivered better results than simple averages of word embedding vectors in document clustering tasks. The results also revealed that k-means clustering provided

the best performance with doc2vec embeddings. The top term analysis was conducted based on a combination of TF-IDF scores and word vector similarities. This method provided a representative set of keywords for a topic cluster. The doc2vec embedding with k-means clustering could successfully recover the latent hashtag structure in Twitter data.

The research of Zhang et al. proposed a novel method to extract topics from bibliometric data [9]. The two innovations presented in the research were a word embedding technique and a polynomial kernel function integrated into a cosine similarity-based k-means clustering algorithm. A word embedding technique was applied in data preprocessing to extract a small set of key features. A polynomial kernel function incorporated with a cosine similarity-based k-means clustering algorithm was implemented to enhance the performance of the topic extraction. The experimental results involving the comparison of the proposed method with k-means, fuzzy c-means, principal component analysis, and topic models demonstrated its effectiveness across both a relatively broad range of disciplines and a given domain. A qualitative evaluation was made based on expert knowledge. Further, several insights for stakeholders were revealed during the qualitative investigation of the similarities and differences between the JASIST, JOI, and SCIM journals.

K-means clustering and decision tree algorithm were integrated in the research of Wang et al. [10]. The research proposed a novel algorithm named "BDTKS" which is a linear multivariate decision tree classifier Binary Decision Tree based on K-means Splitting. The BDTKS algorithm introduced k-means clustering to serve as a node splitting method and proposed the non-split condition; therefore, the proposed algorithm could provide good generalization ability and enhanced the classification performance. Furthermore, the k-means centroid based BDTKS model was converted into the hyperplane-based decision tree; as a result, the speed of the classification process was faster. The experimental results demonstrated that the proposed BDTKS matched or outperformed the previous decision trees.

Chen et al. [11] researched optimal replenishment policies with allowable shortages for a product life cycle. The Nelder-Mead algorithm was employed to find the optimal solution. The purpose of the paper was to determine the optimum number of inventory replenishments, the inventory replenishment time points, and the beginning time points of shortages within the product life cycle by minimizing the total relevant costs of the inventory replenishment system. The proposed problem was mathematically formulated as a mixed-integer nonlinear programming model. There were several numerical examples and corresponding sensitivity analyses presented in the paper. Such examples and analyses were utilized to illustrate the features of the model by employing the search procedure developed in the paper.

3 Methodology

The research proposes a recommendation model based on visitor preferences on commercial websites using a novel algorithm, the "TKD-NM algorithm". The aim of this study is to extend our previous research [12]. The proposed model is divided into three phases as shown in Fig. 1. The first phase is data pre-processing. The second phase

is clustering optimization, and the third phase is webpage suggestion. There were two main tasks to complete in the first phase. The first task was to exclude some data, which were invalid data, missing data, and dependent attributes. Whenever data could not be described, such data was defined as invalid data. Therefore, instances and features that included invalid data were also ignored. In the dataset, some attributes depended on other attributes; therefore, it was necessary to remove dependent attributes from the analysis to reduce computational time. The second task in the first phase was to integrate all related data tables to form a suitable dataset and provide convenience for the next phase.

The second phase of a proposed model is clustering optimization. The objective of this phase is to classify webpages based on their webpage headings. The first step, feature representation, aims to represent each word with its vector. All webpage headings were separated into words using the word_tokenize method from pythainlp.tokenize library. The TF-IDF algorithm was then applied to generate vectors for such words. The next step was to perform clustering using KMeans, and the model was then evaluated by employing the Decision Tree algorithm. There are two approaches in the research. The first approach is the baseline model which implements TF-IDF, KMeans, and Decision Tree only one time and returns the results. In contrast, the second approach is the novel TKD-NM model which reveals the recurring processes of three algorithms; TF-IDF, KMeans, and Decision Tree. Such repetition of the processes is terminated when the highest prediction accuracy and the optimal values of feature vector matrix are obtained. The previously-mentioned optimization algorithm uses a combination of the Nelder-Mead, TF-IDF, KMeans, and Decision Tree algorithms. When the optimization algorithm is complete, it yields optimal word vectors, the target classes of webpages, and the accuracy of the model.

The third phase is webpage suggestion, which provides the recommendation list of webpages regarding visitor preferences. There are two input datasets in this phase. The first input comes from the output of phase 2, which is the relationship between the ID of the webpage and its cluster label. The second input is the relationship between the ID of the visitor and his/her favorite cluster labels. The aim of this phase is to aggregate both datasets, which are the IDs of the webpages in each cluster, the association cluster labels, and the IDs of visitors and their favor cluster labels. The final results are recommendation lists, which are the top ten most possible webpages in the same cluster. The latest webpage based on a published date would be the first in the list, and so on.

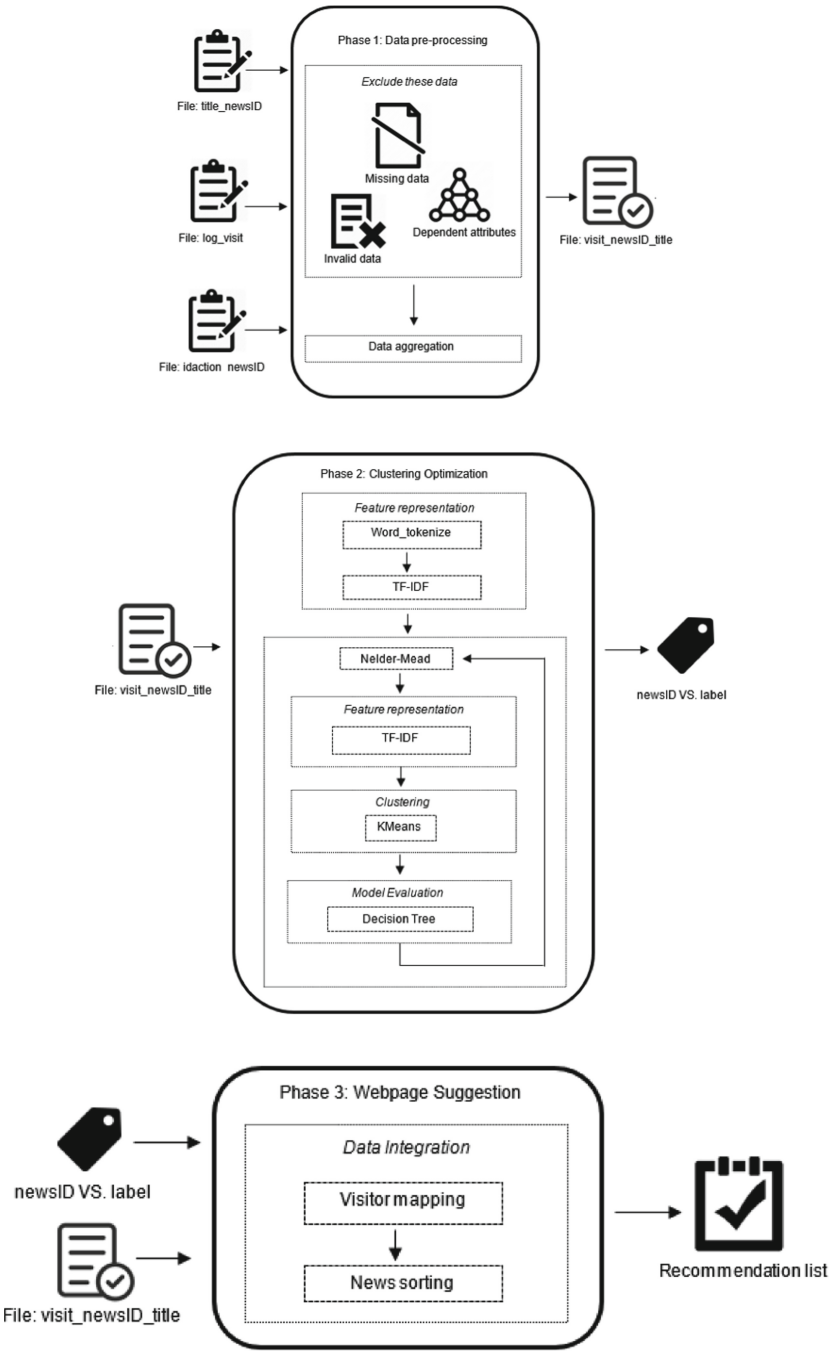

Fig. 1. TKD-NM Algorithm

Algorithm TKD-NM

Input: Feature vector matrix $\boldsymbol{I} \in \Re^{d \times N}$

Output: Optimal feature vector matrix $\boldsymbol{O} \in \Re^{d \times N}$ with assigned class label $\boldsymbol{L} \in \Re^{1 \times N}$

Initialize Ψ to feature vector matrix ***I***.
do
  for i ← 1 **to** N **do**
    for j ← 1 **to** d **do**
      $TFIDF(\Psi)_{i,j} \leftarrow TF(\Psi)_{i,j} * IDF(\Psi)_{i,j}$
    end for
  end for
  $\boldsymbol{K} \leftarrow 6$
  $\boldsymbol{X} = \{x_1, x_2, x_3, \ldots\ldots, x_n\}$; $\boldsymbol{X} \in \Psi$
  $\boldsymbol{V} = \{v_1, v_2, v_3, \ldots\ldots, v_K\}$
  $\boldsymbol{S} = \{s_1, s_2, s_3, \ldots\ldots, s_K\}$
  Initialize ***K*** centroids as $v_1, v_2, v_3, \ldots\ldots, v_K$ at random from Ψ.
  do
    for i ← 1 **to** ***K*** **do**
      $pv_i = v_i$
    end for
    $D(V) = \sum_{i=1}^{K} \sum_{j=1}^{c_i} (||x_i - v_i||)^2$
    for i ← 1 **to** K **do**
      for j ← 1 **to** c_i **do**
        $s_i \supset x_j$
      end for
    end for
    for i ← 1 **to** K **do**
      for j ← 1 **to** c_i **do**
        v_i = **Mean**(s_j)
      end for
    end for
  while $(pv_1 \neq v_1 \,||\, pv_2 \neq v_2 \,||\, \ldots\ldots\,||\, pv_K \neq v_K)$
  $\boldsymbol{L} \leftarrow \boldsymbol{S}$
  Assign Ψ to root node.
  max_depth = 0
  do
    for i ← 1 **to n do**
      E_i(root node) = $-p_i log_2 p_i$
      IG_i(root node) = E_i(root node) - $\sum_{j=1}^{m} E_j$ (root $node_j$)
    end for
    $a \leftarrow$ attribute with smallest E(root $node_a$) and largest IG(root $node_a$)
    Split root node by ***a***
    max_depth++
    assign each subset to root node
  while (max_depth < 27**)**
while (prediction accuracy has not converged yet**)**
Assign Ψ to optimal feature vector matrix ***O***.

Algorithm has sketched out the process of forming the TKD-NM algorithm. The input of the algorithm is the feature vector matrix of webpage headings while the output is the optimal feature vector matrix from the TKD-NM algorithm. Lines 3–7 represent the webpage headings with feature vectors. In this experiment, the number of clusters is 6. $\boldsymbol{X}$ is the set of data points. $\boldsymbol{V}$ is the set of centroids. $\boldsymbol{S}$ is the set of data points in each cluster. Line 17 calculates the distance between each data point and its nearest centroid. $\boldsymbol{c_i}$ is the number of data points in $\boldsymbol{i^{th}}$ cluster. Lines 18–22 assign each data point x to the closet centroid. Lines 23–27 compute the centroids for the clusters by taking the average of the all data points that belong to each cluster. Line 34 calculates the entropy of each attribute. Line 35 calculates the information gain of each attribute. Line 37 selects an attribute which has the smallest entropy value and the largest information gain value.

4 Experimental Results

Three datasets were collected from back-end websites. The first dataset was the headings of webpages and their IDs. The number of instances for the first dataset was 103,860. The second dataset were data of visits for nine months, such as the ID of the visitor, the ID of action, and the visitor location. There were 53,054 instances with 25 features. The third dataset comprised the ID of action and the ID of the webpage heading, and there were 4,997 instances. The number of instances was reduced to 14,840 instances after missing values, invalid data, and dependent attributes were removed. The novel algorithm, TKD-NM, was applied to yield recommendation lists of webpages. The elbow method was adopted to obtain the optimal k as shown in Fig. 2. To conclude from Fig. 2, the experiment for clustering would consider a value for k between 3 and 13.

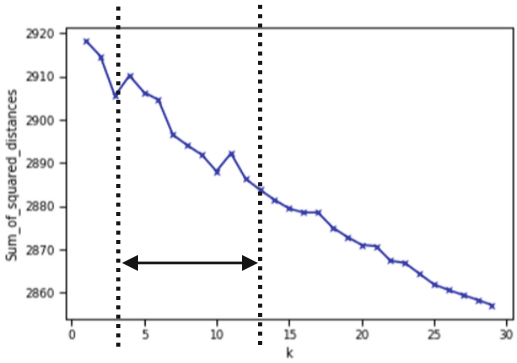

Fig. 2. The elbow method for optimal k between 1 and 29

Another three machine learning techniques, namely KNN, Decision Tree, and Multi-Layer Perceptron (MLP), were utilized to determine a suitable number of clusters. The experiment of KNN was conducted with the variation of k; however, the optimal value of k was 300. When Decision Tree was applied, the variation of depth was evaluated and it was revealed that the optimal depth was 27. MLP was conducted with various

numbers of hidden layers and hidden nodes. The optimal number of hidden layers was 2 with 50 hidden neurons. The prediction accuracy of each machine learning technique was plotted as shown in Fig. 3. To conclude from Fig. 3, Decision Tree yielded high performance results when k was between 3 and 7. Therefore, the research selected to perform clustering based on six groups and employed the Decision Tree algorithm to evaluate both the baseline model and the novel TKD-NM model.

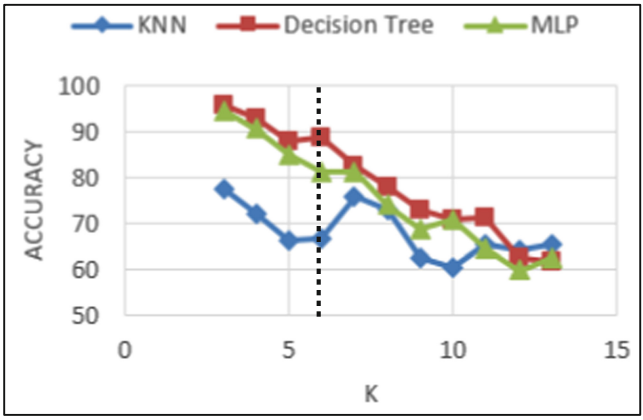

Fig. 3. The prediction accuracy of KNN, Decision Tree, and MLP

This research was constructed based on two approaches. The first approach was a baseline method, which was performed by using the TF-IDF, KMeans, and Decision Tree algorithms. TF-IDF was adopted for feature vectorization of webpage headings. KMeans was applied for webpage heading clustering while Decision Tree was employed to evaluate the prediction model. These three algorithms were implemented once and returned the output as soon as they were completed. The second approach was the novel algorithm, TKD-NM. This proposed method also utilized the TF-IDF, KMeans, and Decision Tree algorithms, but it required an extra algorithm which was Nelder-Mead. The objective of the Nelder-Mead algorithm was to obtain the optimal values of the feature vector matrix. When the optimal values of the feature vector matrix were reached, the highest accuracy of the prediction model would be generated. An example of webpage headings and their separated words is presented in Fig. 4. An example of a feature vector matrix is shown in Fig. 5. The experimental results of the baseline model and the TKD-NM model are shown in Table 1 and reveal that the TKD-NM model yields higher prediction accuracy than the baseline model. The prediction accuracy of TKD-NM model is approximately 97.31% while the baseline model offers 88.87% prediction accuracy. Although the TKD-NM model outperformed the baseline model, the processing time of the TKD-NM model was slower than the baseline model because the complexity of the proposed model was time-consuming.

An example of a recommendation list for one visitor is shown in Table 2. The top five webpages were retrieved from webpages in the same cluster and were sorted by date. The first order of the recommendation list is the latest uploaded webpage.

newsID	title	word
98380	ดอกไม้ยอดฮิตปากคลองตลาดพร้อมความหมาย	['ดอกไม้', 'ยอดฮิต', 'ปากคลอง', 'ตลาด', 'พร้อม...
107246	เก็บภาษีความหวานรอบสองเริ่มแล้วตั้งแต่วันนี้กย	['เก็บภาษี', 'ความหวาน', 'รอบ', 'สอง', 'เริ่ม'...
107685	เรืองย่อมนตรามหาเสน่ห์ตอนที่ไออกอากาศสิงหาคม	['เรืองย่อ', 'มน', 'ตรา', 'มหา', 'เสน่ห์', 'ต...
107692	ผลสลากกินแบ่งรัฐบาลงวดวันที่สค	['ผล', 'สลากกินแบ่งรัฐบาล', 'งวด', 'วันที่', '...
107711	แกงส้มโต้ดึงวันเดอร์เฟรมเข้าค่ายเพลง	['แกงส้ม', 'โต้', 'ดึง', 'วัน', 'เด', 'อร์', '...

Fig. 4. Webpage headings and separated words

```
  (0, 1586)     0.22508362159088907
  (0, 1439)     0.1616932486612373
  (0, 1413)     0.17828202668859255
  (0, 1228)     0.1772136763321552
  (0, 873)      0.24695843579061105
  (0, 797)      0.3519401521176865
  (0, 639)      0.3300653379179645
  (0, 605)      0.3145449102470466
  (0, 527)      0.2397544265057669
  (0, 267)      0.3300653379179645
  (0, 179)      0.3519401521176865
  (0, 111)      0.3519401521176865
  (0, 93)       0.23048311717415768
```

Fig. 5. Feature vector matrix of each word

Table 1. Experimental results

Algorithm	Prediction accuracy (%)	No. of clusters
Baseline model	88.87	6
TKD-NM model	97.31	6

Table 2. Recommendation lists for one visitor

Webpage	Rank	Cluster No.
#####.com/sport/news/102749	1	3
#####.com/news/ประเด็นร้อน/103209	2	3
#####.com/news/ประเด็นร้อน/102124	3	3
#####.com/sport/news/87608	4	3
#####.com/sport/news/39847	5	3

5 Conclusion and Suggestions

The research proposed a recommendation model based on visitor behaviors on commercial websites using the TKD-NM algorithm, which is the combination of the TF-IDF, KMeans, Decision Tree, and Nelder-Mead algorithms. The elbow method was adopted to investigate the optimal value of k. Decision Tree, KNN, and MLP were also applied to evaluate the algorithm. The experimental results revealed that the optimal number of clusters was between 3 and 7. However, the research conducted six clusters of webpages defined as 0 to 5. The Decision Tree algorithm was applied to evaluate both the baseline model and the TKD-NM model. The prediction accuracy of the proposed model was higher than the baseline method; the TKD-NM algorithm yielded 97.31% while the baseline algorithm provided 88.87% accuracy. However, other adaptations and experiments could be employed in future research. Since the proposed model might be time-consuming when processing a large amount of data, future work could involve deeper analysis of particular mechanisms to leverage the speed of the optimization process.

Acknowledgments. This research is financially supported by the Department of Computer Science, Faculty of Science, Kasetsart University. Moreover, the authors would like to express their gratitude to an anonymous company which cannot be mentioned because of confidentiality. It is appreciated that the business data provided by this selected company is sensitive and will not be disclosed or used for any purpose other than for research.

References

1. Steve Jobs Quote: What Customers Can Tell You https://www.entrepreneurshipinabox.com/quotes/steve-jobs-quote-what-customers-can-tell-you/. Accessed 13 Feb 2021
2. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. **17**, 734–739 (2005)
3. Kim, D., Seo, D., Cho, S., Kang, P.: Multi-co-training for document classification using various document representations: TF-IDF, LDA, and Doc2Vec. Inf. Sci. **477**, 15–29 (2019)
4. Padurariu, C., Breaban, M.: Dealing with data imbalance in text classification. ScienceDirect **159**, 736–745 (2019)
5. Song, Y., Wu, S.: Slope one recommendation algorithm based on user clustering and scoring preferences. ScienceDirect **166**, 539–545 (2020)
6. Portugal, I., Alencar, P., Cowan, D.: The use of machine learning algorithms in recommender systems: a systematic review. Expert Syst. Appl. **97**, 205–227 (2018)
7. Erra, U., Senatore, S., Minnella, F., Caggianese, G.: Approximate TF-IDF based on topic extraction from massive message stream using the GPU. Inf. Sci. **292**, 143–161 (2015)
8. Curiskis, S., Drake, B., Osborn, T., Kennedy, P.: An evaluation of document clustering and topic modelling in two online social networks: Twitter and Reddit. Inf. Process. Manag. **57**, 102034 (2020)
9. Zhang, Y., et al.: Does deep learning help topic extraction? A kernel k-means clustering method with word embedding. J. Informetr. **12**, 1099–1117 (2018)
10. Wang, F., Wang, Q., Nie, F., Li, Z., Yu, W., Ren, F.: A linear multivariate binary decision tree classifier based on K-means splitting. Pattern Recogn. **107**, 107–120 (2020)

11. Chen, C., Hung, T., Weng, T.: Optimal replenishment policies with allowable shortages for a product life cycle. Comput. Math. Appl. **53**, 1582–1594 (2007)
12. Chaipornkaew, P., Banditwattanawong, T.: A recommendation model based on user behaviors on commercial websites using TF-IDF, KMeans, and Apriori algorithms. In: Meesad, P., Sodsee, D.S., Jitsakul, W., Tangwannawit, S. (eds.) Recent Advances in Information and Communication Technology 2021. IC2IT 2021, pp. 55–65. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-79757-7_6