Pertanika J. Sci. & Technol. 34 (1): 471 - 497 (2026)

SCIENCE & TECHNOLOGY

Journal homepage: http://www.pertanika.upm.edu.my/

An Enhanced Sentiment Score Prediction in the Collaborative Filtering Algorithm through a Hybrid-Dynamic-Topic-Sentiment User-based KNN (HDTS-uKNN)

Shuai Shuai Wang and Noor Farizah Ibrahim*

School of Computer Sciences, Universiti Sains Malaysia, 11800 Gelugor, Pulau Pinang, Malaysia

ABSTRACT

Currently, social media generates a large amount of content every day, with relevant content including the experience of product performance, price, and branding, which can assist retailers in understanding the change of consumers' sentiment toward a certain type of product and provide recommendations for consumers to purchase related products. Although different existing recommendation systems have recommended products through users' ratings, the recommendation systems generally neglect the time factor, contain sparse data, and lack the ability for semantic understanding. The current study proposed an enhanced collaborative filtering algorithm, namely the Hybrid-Dynamic-Topic-Sentiment user-based K-Nearest Neighbors (HDTS-uKNN), based on the topic model, sentiment analysis, and time decay function. The experimental results revealed that the proposed algorithm demonstrated enhanced performance compared to other baseline algorithms on the Twitter dataset, with the lowest mean absolute error (MAE) score of 0.1003. The proposed collaborative filtering algorithm not only aided in resolving the data sparsity issue but also utilised the dynamic hybrid similarity method to contribute to more refined product and service recommendations.

Keywords: Collaborative filtering, LDA, sentiment analysis, time decay factor

ARTICLE INFO

Article history:
Received: 08 May 2025
Accepted: 21 October 2025
Published: 25 February 2026

DOI: https://doi.org/10.47836/pjst.34.1.21

E-mail addresses:
wangshuaishuai@student.usm.my (Shuai Shuai Wang)
nfarizah@usm.my (Noor Farizah Ibrahim)
* Corresponding author

INTRODUCTION

The rapid development of the internet has fostered various interactions and communications on social media communities, which has generated a large amount of content and rendered it difficult for individuals to search for relevant information. To address the difficulty, an efficient recommendation system may assist

e-ISSN: 2231-8526

in filtering out unrelated information in social media and e-commerce. Among e-commerce companies, the recommendation system plays the role of a virtual expert by discovering the user's taste and preference from a large amount of user data, and customises the recommendation system to fulfil the needs of users and enterprises (Chehal et al., 2021; Song et al., 2019). Collaborative filtering (CF) is the most common recommendation method among different recommendation systems (RSs), which has been widely employed by Amazon and other e-commerce domains to recommend items based on the preferences of the target user and other users, such as past behaviours, ratings, and comments about items. Hence, CF-RSs can be further categorised into two primary types, namely user-based and item-based (Rizkallah et al., 2021), with the user-based method employed in the current study.

Traditional memory-based CF-RSs utilised traditional similarity methods (e.g., Cosine, Pearson, and Adjust-Cosine) to calculate the similarity of users based on the rating matrix or score matrix. Nevertheless, the similarity values between users have frequently led to inaccuracy in the prediction process when the score vectors are sparse. Furthermore, traditional CF-RSs constantly considered only rating features to provide recommendations while ignoring users' reviews and the change of users' preferences on items over time (Liu & Zhao, 2023; Sun et al., 2019). The traditional CF-RS also continuously assigned the same weight to the items, which impacted the similarity values between the users. While numerous companies have developed a variety of techniques and business strategies to capture different content, the lack of pertinent techniques to analyse large-scale unstructured data has demonstrated an urgent need for more robust measurement and analysis methods (Chen et al., 2023; Talhaoui et al., 2018; Wedel & Kannan, 2016). Simultaneously, the existing data modelling methods, including Latent Dirichlet Allocation (LDA) and sentiment analysis, are generally utilised for text analysis, text topic classification, and text sentiment identification (Jelodar et al., 2019; Kumar et al., 2020).

Multiple existing studies have applied topic modelling, sentiment analysis, and the time decay factor to improve the performance of RSs. Notwithstanding, the performance validation has mostly been conducted on structured benchmark datasets, such as MovieLens, Last.fm, or Kaggle Amazon Reviews. The interaction behaviour in the datasets is frequently restricted to specific domains, such as MovieLens being limited to movies and Last.fm to the music domain. The datasets are organised data, which are not the same as the reviews that users have posted on social media. Contrastingly, Twitter data is more realistic, as the timestamps and data are real-time expressions performed by users in the current situations, which reflect personal emotions. Therefore, the Twitter dataset can aid e-commerce platforms in instantaneously understanding user feedback and adjusting recommendation strategies accordingly. Meanwhile, the deployment environment resources are limited for small- and medium-sized firms (SMEs) (Yuwono et al., 2024). A lightweight and interpretable approach may provide an intuitive predictive result for platforms and merchants with limited resources, which may allow more opportunities to gain merchants' trust in the system (Chatti et al., 2024; Zhang & Chen, 2020) rather than relying on black-box predictions that merchants cannot observe the specific user features that have led to the recommendation results. Therefore, exploring a lightweight and interpretable approach for SMEs can be valuable.

Data sparsity, semantic ignorance, and temporal shifts are generic and highly acknowledged challenges in RSs, and the issues are particularly prominent on Twitter. Particularly, tweets are short texts (Kaveri & Maheswari, 2019), and users' postings are uneven, in which certain users post a large number of tweets with rich information, whereas other users post only several tweets with limited and noisy information. The circumstance may lead to data sparsity and more challenges for traditional methods in stably capturing user preferences. Accordingly, the primary objective of the present study was to validate the applicability and performance of the proposed hybrid method on the Twitter data to provide support for the future applicability of the method in utilising the Twitter dataset as a supplementary signal on real e-commerce data. Specifically, a lightweight and interpretable approach, which integrated the time decay function into the topic and sentiment analysis models to reduce the influence of outdated scores or noise on the similarity calculation process and capture the user preferences from the short contents, was proposed.

A hybrid similarity method, which was proposed in the current study, combined the semantic similarity method and traditional similarity method based on the results of the dynamic topic and sentiment analysis models for similarity calculation. The proposed Hybrid-Dynamic-Topic-Sentiment user-based K-Nearest Neighbors algorithm (HDTS-uKNN) employed different analytics to predict the sentiment score of the consumers to enhance dynamic CF for retail service management. The current study is structured as follows. Section 2 describes the related work of recommendation algorithms, time decay factor, topic model, and sentiment analysis. Section 3 delineates the design and components of the proposed algorithm. Section 4 presents and discusses the experimental results of the proposed algorithm and baseline algorithms on the Twitter data. Section 5 summarises the experimental results and discusses future research directions.

RELATED WORK

Different RSs are designed to more effectively handle the issue of information overload and provide users with personalised recommendations. Since the mid-1990s, a variety of recommendation system technologies have been developed to provide convenience for businesses, governments, education, and other domains. Generally, RSs can be classified into three categories, namely (CF), content-based filtering (CBF), and hybrid filtering (HF). The first category, namely CF, is one of the most common techniques that apply in

RSs, which performs recommendations based on similar historical hobbies between the target user and other users, and CF is implemented through memory-based and model-based methods. The memory-based CF method can be divided into user-item filtering and item-item filtering. Meanwhile, the model-based CF method performs recommendations through deep learning, matrix factorisation, and clustering algorithms.

Traditional memory-based CF did not require model training, which rendered it more susceptible to data fluctuations (Gong et al., 2009). Increasing research attention on CF systems has propelled RSs to demonstrate significant progress in terms of personalisation and accuracy. Researchers have also explored temporal dynamics, sentiment factors, and semantic information in CF-based RSs. Specifically, the time decay factor assists RSs in capturing user preference variation in the long term and allows RSs to rely more on the current user Behavioural data. For instance, Chen et al. (2021) proposed a CF with a time decay factor to capture user preference variation via user ratings as a proxy for user preferences. The results uncovered that CF with time decay demonstrated enhanced performance compared to CF that only considered the one-time window. In addition, CF could incorporate the time decay factor, cosine similarity, and coupled tensor factorisation techniques to capture changes in user preferences in the long term (Tahmasbi et al., 2021) by considering the user preferences dynamically, with the personalised time decay factor significantly improving the accuracy of the systems.

Jain et al. (2022) proposed a CF with a time decay factor and other models, such as a deep neural network and matrix factorisation, to address the data sparsity and user preference change in the long term, due to the sparsity issue of the RSs. The results revealed that the time decay factor could aid CF in managing user preference changes, and the time decay functions determined the appropriate time weights, which enabled the system to focuss more on the user's recent ratings. Nonetheless, the present study did not consider the semantic information. Furthermore, user interaction data or preferences are vital to the system at varying levels, and different phases of interaction data should not be treated to the same degree. Dividing the timeline into different stages is also one of the approaches to improving system performance. The days, weeks, and months can be utilised as the dividing line for different time weights (Rafailidis & Nanopoulos, 2016; Tahmasbi et al., 2021).

The LDA is a probabilistic topic model aimed at representing the potential topic distribution of user historical Behaviour. Wilson et al. (2014) proposed CF with LDA to account for the potential relationship between users and items, although the time decay factor was not considered, and a suitable number of topic combinations was also required. Moreover, the LDA could facilitate the system to understand the contents between individuals, such as extracting the topic from the interaction data that users were interested in, despite the topic model not understanding the meaning of the words (Jelodar et al., 2019). Tang et al. (2020) and Jiang et al. (2023) also utilised the topic model LDA to extract the movie and course features from the description, and the results demonstrated that CF with the LDA outperformed CF only on the ratings, CF with random method (RM), knowledge topic vector-based method (KTVM), and course vector- based method (CVM). Nonetheless, the studies did not consider the time factor, extending the data embedding algorithm Graph Generative Adversarial Nets (GraphGAN), and the dynamic change.

Sentiment analysis is a technology that can assist in identifying and categorising individuals' emotions, perspectives, and attitudes toward products or services, and CF with sentiment analysis can be utilised to identify the opinions of users' comments more accurately. Sun et al. (2019) employed HowNet to analyse the users' reviews, and the results showed that the CF with sentiment analysis could aid in capturing more information compared to only utilising ratings. Notwithstanding, the study did not consider the time decay factor and also suffered from cold-start and data sparsity issues when no comments were from new users. Additionally, Abbasi-Moud et al. (2021) and Asani et al. (2021) proposed a personalised RS by utilising sentiment analysis (Sentiwordnet) to offer suggestions on tourists and foods, and the sentiment analysis leveraged the hidden feelings from users' comments to improve the accuracy of the RSs. Another sentiment analysis technique (LinearSVC) was also applied in the drug RS to detect the users' reviews of drugs and reduce the specialists' workload (Garg, 2021). The results uncovered that the LinearSVC on TF-IDF outperformed the Perceptron on Bow, LGBM on Word2Vec, and Random Forest on manual features. Furthermore, Kumar et al. (2020) proposed CF with sentiment analysis (Valence Aware Dictionary and Sentiment Reasoner or VADER), cosine similarity, and a neighborhood-based prediction method, which utilised tweets and ratings to improve the accuracy of the RS. The results uncovered that the execution of VADER was swifter compared to Naive Bayes analyser and TextBlob, although the dynamic paradigm of recent movies was not considered.

Deep learning, which is a novel subfield of machine learning, and both supervised learning and unsupervised learning methods can be applied to RSs. Deep learning can assist in enhancing RS performance by leveraging the feature representation learning capabilities to learn complex relationships (non-linear and non-trivial) hidden in user-item interaction data (Jangid & Kumar, 2025; Zhang et al., 2019). Recent studies have also applied deep learning for enhancing recommendation quality. Devika and Milton (2025) proposed an RS leveraging sentiment analysis and hybrid deep learning models. The proposed method conducted sentiment analysis, namely TextBlob, to extract the polarity from the Amazon Book Review dataset before utilising the transformer (BERT) to generate relevant word embeddings. The proposed ensemble model, which combined the predictions of three hybrid model architectures, outperformed the performance of three distinct hybrid model architectures (CNN-LSTM-GRU, CNN-BiLSTM-GRU, and LSTM-CNN).

Chen et al. (2025) proposed a hybrid recommendation method employing deep learning (DNN) and CF. The proposed method applied digital encoding to represent user feature vectors (age, gender, or browsing history) before extracting product feature vectors (text description and images) through a transformer (BERT) and ResNet embedding. The user and item feature vectors were concatenated as the input of DNN to predict the user's score about the product, and the final prediction score for a user about a product was predicted through a weighted average of scores of DNN and CF. The results demonstrated that the DNN-CF outperformed DNN-only and CF-only. Nevertheless, the methods required profound computational resources owing to the complex structure. Additionally, the training and inference processes were highly memory-intensive and dependent on the quality of the training data. Meanwhile, Zhu and Samsudin (2024) presented a sentiment analysis method to obtain users' opinions from movie reviews through spatialised word embedding, Bi-LSTM, and the attention mechanism. While the method could capture more accurate user opinions from long sentences compared to other baseline methods (Bi-LSTM or Bi-LSTM + Attention), the running time and computational resource consumption were higher.

Another approach is a transformer, which is a deep learning model architecture that leverages global dependencies. In contrast to traditional deep learning methods (CNN and RNN), the transformer highly relies on the self-attention layers (mechanism) and can simultaneously compute relationships between vectors at different positions globally. Vaswani et al. (2017) incorporated the positional encoding to inject the relative position of words in the sequence, which aided in enabling the transformer model to perceive sequence order. The drawback was that the conventional positional encoding did not account for differences across different time periods and user behaviours (Zhou et al., 2023), as CNN and RNN required either multi-layer stacking or step-by-step propagation to compute global dependencies (relationships between vectors at different positions). The advantage of CNN and RNN is capturing the sequence information during stacking and propagation, whereas the limitation is that CNN and RNN compute global dependencies indirectly and suffer from information decay. Although current deep learning-based models, including CNN, LSTM, GRU, and DNN, and transformer-based models, such as BERT, dual transformers, and dynamic weight transformers, have enhanced the performance of RSs, the methods have constantly exhibited several limitations, namely (1) highly dependent on the quality of the training data, (2) limited interpretability of the recommendation process, (3) requiring substantial computational resources, and (4) highly memory-intensive training and inference processes.

Sentiment analysis (VADER), LDA, and time decay function have been applied in other recommendation system studies (Chen et al., 2021; Kumar et al., 2020; Na et al., 2021). Nonetheless, existing studies have either individually examined the performance of CF with a time decay function, CF with LDA and a time decay function, or hybrid RS with VADER on structured benchmark datasets, such as Movielens. The studies have not explored the predictive effects of LDA with the time decay function, VADER with the time decay function, and the Hybrid similarity method for CF on short text, such as tweets.

Hence, the current study proposed an integrated CF framework integrating the LDA, VADER, time decay function, and hybrid similarity method to predict the users' preferences, seeking to explore the applicability and combinational effects of the models specifically on the unstructured real-time dataset like social media data. In the proposed method, the temporal awareness where recent interactions are weighted more than outdated one were explored to align with dynamic structure of social media data. The proposed method, namely HDTS-uKNN, emphasised combining the LDA-time decay function, VADER-time decay function, and hybrid similarity method in a real-world Twitter environment to provide a dynamic, lightweight, and interpretable RS for SMEs and able to mitigate the limitations of single-similarity approaches.

METHODOLOGY

The current section introduces the framework of the proposed HDTS-uKNN. The hybrid similarity calculation and prediction process are discussed after the derivation process of the dynamic LDA model and dynamic sentiment analysis. In addition, the fundamentals of LDA and sentiment analysis are delineated.

The Basic LDA

Figure 1 depicts the process of LDA modelling corpus *D*, with the LDA assuming that corpus *D* consists of *M* documents, and each document (d) contains *N words*. Specifically, α, β are vector hyperparameters, where in α is the scale parameter that controls the document-topic distribution, and each document d contains more topic numbers when α is bigger. Meanwhile, β is the topic parameter that controls topic-words distribution, with each topic t including more different words when β is bigger.

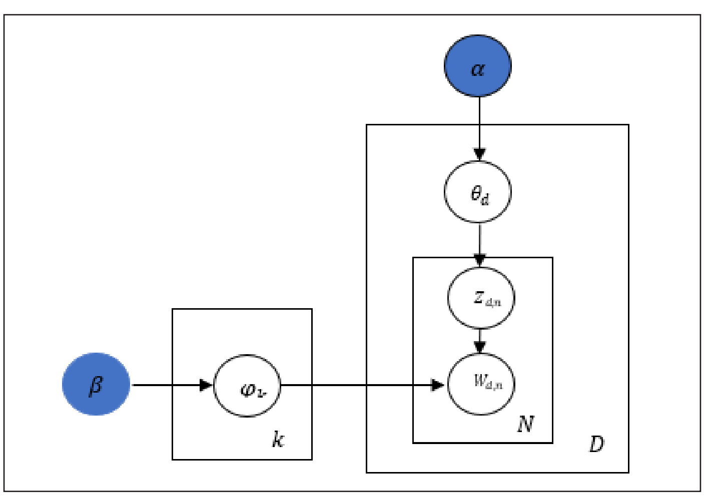

Figure 1. Latent Dirichlet allocation model

For the topic distribution θ_d of each document d ($d \in 1,\dots,M$) in the corpus D, the formula can be understood as choosing a topic distribution θ_d from the Dirichlet distribution with hyperparameter α, as indicated in Equation 1:

$$\theta_d = \text{Dirichlet}(\vec{\alpha}) \quad [1]$$

For the word distribution φ_k of each topic number $Z_{d,n}(k \in \{1,\dots,K\})$ in document d, the formula can be understood as choosing a word distribution φ_k from the Dirichlet distribution with hyperparameter β, as indicated in Equation 2:

$$\varphi_k = \text{Dirichlet}(\vec{\beta}) \quad [2]$$

For the allocation topic number $Z_{d,n}$ of each word $W_{d,n}$, each word $W_{d,n}$ based on the topic distribution θ can be allocated to the topic number $Z_{d,n}(Z_{d,n} \in \{1,\dots,k\})$, as indicated in Equation 3:

$$Z_{d,n} = multi(\theta_d) \quad [3]$$

For the distribution of the n^{th} word $W_{d,n}$ belong to topic number $Z_{d,n}$, the distribution is represented as Equation 4:

$$\text{P}(\text{W}_{\text{d,n}}) = \text{multi}(\varphi_{Z_{\text{d,n}}}) \quad [4]$$

For deriving the optimal hyperparameters α, β, the optimal hyperparameter values α, β can be inferred by observing the maximised probability value of corpus D via Equation 5, wherein words $W_{d,n}$ are the observed variables according to the word frequency that can be observed in the documents, with the topic distribution θ_d and word distribution φ_k as the latent variables as shown in Equation 5:

$$\text{p}(\text{D}|\alpha,\beta) = \prod_{d=1}^{M} \int \text{p}(\theta_d|\alpha) \times \left(\prod_{n=1}^{N_d} \sum_{z_{dn}} \text{p}(z_{dn}|\theta_d)\text{p}(w_{dn}|z_{dn},\beta) \right) d\theta_d \quad [5]$$

VADER Sentiment Analysis

The sentiment-lexicon-based approach, namely the VADER natural language processing tool, was employed to scrutinise sentiments toward a certain topic based on the lexical features, including words. In the process of calculating the sentiment of texts, each word was provided with a sentiment score according to the polarity of the word, either positive, negative, or neutral, and the overall sentiment polarity of tweets was obtained through the weighted average of the sentiment polarities of each word. The polarity of each word and the probability of positive, negative, neutral, or compound are according to the rules presented in Table 1.

Table 1
Rules of polarity

```
def sentiment(x):
if x[“compound”] >=0.5:
return “positive”
elif x[“compound”] <=0.05 and x[“compound”]>=-0.05:
return “neutral”
elif x[“compound”] <=-0.05:
return “negative”
```

Note. “x” represented each text, and the sentiment value of x[“compound”] ranges from [-1, 1]

As demonstrated in Equation 6 or the normalisation function, “x” represented each text, and the sentiment value of each text ranges from [-1, 1], as VADER internally includes a normalisation process as shown in Equation 6:

$$x[\text{"compound"}] = \frac{x["compound"]}{\sqrt{x["compound"]^2 + 15}} \quad [6]$$

Generating the HDTS-uKNN Algorithm

The topic model approaches applied by previous scholars did not process the documents of the corpus in chronological order, and the documents in the corpus could be swapped with each other. The traditional topic model LDA assigned the same weight to different documents while ignoring the impact of time. Changes in users’ interests and hobbies might be neglected by the topic model, as users’ preferences might vary from week to week, and interests might change over time. Therefore, the accuracy of the recommendation system could be influenced by the documents at different times that are not distinguished based on time sequence (Na et al., 2021). While the sentiment score of each tweet could be obtained by the weighted average of the compound value of each word through VADER, traditional VADER did not consider the effect of the time factor, and the sentiment value might be less reliable in the RS when the tweet was longer. To address the issues, the current study incorporated the time decay function into the proposed topic model (t-LDA) and sentiment analysis model (t-SENTIMENT) to capture the changes of interest over time, in which the nearest documents were offered a high weight, and the weight gradually decreased as moving further away from the present time. Figure 2 illustrates the framework of the proposed HDTS-uKNN, which comprises three primary phases, namely the matrix forming by the HDTS model (consisting of t-LDA and t-SENTIMENT) in Phase 1, the hybrid similarity calculation in Phase 2, and the sentiment score prediction by uKNN in Phase 3.

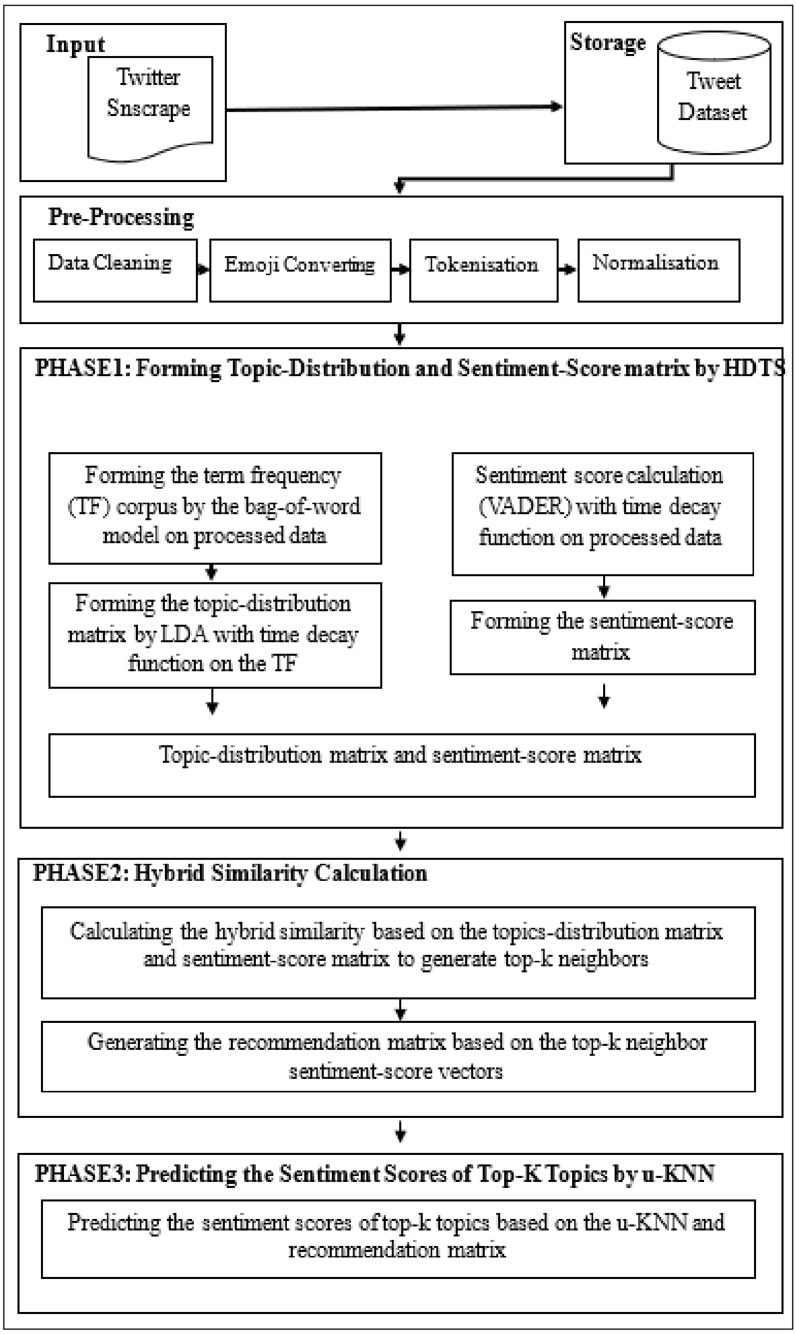

Figure 2. Framework of HDTS-uKNN

Matrix Forming by HDTS

The bag of words was employed to transform the unstructured dataset (tweets) into the language term frequency (TF) that the computer could understand by forming the topic-distribution matrix through t-LDA. Subsequently, the topic LDA model was developed to consider the creation time of each document, and the corpus was divided into T time slices in chronological order. Each time slice contained seven days of documents, and the documents under each time slice could be represented as $D = \{d_1, d_2 \ldots d_T\}$. Furthermore, t_r is the recent time of the documents, and t_i is the past extraction time of the documents. The time weight (r) expression is represented as Equation 7:

$$\text{days\_diff} = (\text{t}_r - \text{t}_\text{i}) \cdot \text{days}$$

$$\text{weeks\_diff} = \text{math.ceil}\left(\frac{\text{days\_diff}}{7}\right) \quad [7]$$

$$\text{r}(\text{t}_\text{i}) = \text{rate}^{\text{weeks\_diff}}$$

The rate is a fixed value of 0.95, days is a Python library for calculating the number of days, and $math.ceil(x)$ is a Python library for rounding up x to the nearest integer. The $\text{TF}_\text{t}^\text{d}$ of the current document is expressed as Equation 8:

$$\text{TF}_\text{t}^\text{d} = \text{TF}_d \cdot \text{r}(\text{t}) \quad [8]$$

The distribution φ_t of the current topic is expressed as Equation 9:

$$\varphi_\text{t}^\text{k} = \varphi_\text{k} \cdot \text{r}(\text{t}) \quad [9]$$

The distribution of the document θ_d is influenced by the time delay rate, and θ_d is expressed as Equation 10:

$$\theta_t^d = \theta_d \cdot r(t) \quad [10]$$

The process of generating the distribution of a document in time segment t is represented as Algorithm 1.

The time decay function (Equation 11) was added to the VADER sentiment rule in the current study by forming the sentiment-score matrix through t-SENTIMENT to manage the time attributes of the tweets. The sentiment calculation rule after adding the time decay function is portrayed in Table 2.

Algorithm 1: LDA with time decay factors algorithm

Input: Corpus D, hyperparametres $\boldsymbol{\alpha}, \boldsymbol{\beta}$, optimal topic number K

Output: Latent variables $\boldsymbol{\varphi_t}$ and $\boldsymbol{\theta_d}$

For each document d:

If time slice T ==1:

$$\mathbf{TF_t^d} = \mathbf{TF}_{\boldsymbol{d}}$$

Else:

Calculate $\mathbf{TF_t^d} = \mathbf{TF}_{\boldsymbol{d}} \cdot \mathbf{r(t)}$ according to Equation 8

For each topic k:

Calculate $\boldsymbol{\varphi_t} \sim \boldsymbol{Dir}(\boldsymbol{\beta})$

For each document d:

Calculate $\boldsymbol{\theta_t^d} \sim \boldsymbol{Dir}(\boldsymbol{\alpha})$

$$\text{days\_diff} = (\text{last\_date} - \text{extract\_date}) \cdot \text{days}$$

$$\text{weeks\_diff} = \text{math.ceil}(\frac{\text{days\_diff}}{7}) \quad [11]$$

$$\text{decay\_rate} = \text{decay\_rate}^{\text{week\_diff}}$$

The decay_rate is a fixed value of 0.95, last_date is the recent date of the documents, extract_date is the extraction date of the documents, days_diff is the number of days between the current date and the extraction date, and weeks_diff is the number of days converted to weeks.

Table 2
Rule of polarity with the adding time decay function

def sentiment(x):
if x[“compound”] · decay_rate >=0.05:
return “positive”
elif x[“compound”] · decay_rate <=0.05 and x[“compound”] · decay_rate >=-0.05:
return “neutral”
elif x[“compound”] · decay_rate <=-0.05:
return “negative”

Note. “x” represented each text, and the sentiment value of each text ranges from [-1, 1]

Hybrid Similarity Calculation

A key step in the CF-RSs is to calculate the similarity between users through similar calculation methods. The top k neighbors with high to low similarity values were selected based on angles or correlations between other users and the target user. The similarities of angle and correlation between users could be calculated via traditional similarity methods and semantic similarity methods. The traditional similarity methods included cosine-based similarity, Pearson's correlation coefficient, and adjusted cosine (AC) Equations 12, 13, and 14, which were employed to calculate the similarity based on the scores and ratings. The rating information in the present study was not included in the tweet content. The t-SENTIMENT model was utilised to generate the user-topic sentiment scores matrix for all users.

$$\mathrm{PCC}(u_t, u_n) = \frac{\sum_{i \in I_t \cap I_n} (u_{t,i} - \bar{u}_t) \cdot (u_{n,i} - \bar{u}_n)}{\sqrt{\sum_{i \in I_t \cap I_n} (u_{t,i} - \bar{u}_t)^2} \cdot \sqrt{\sum_{i \in I_t \cap I_n} (u_{n,i} - \bar{u}_n)^2}} \quad [12]$$

$$\mathrm{Cosine}(u_t, u_n) = \frac{\sum_{i \in I_t \cap I_n} u_{t,i} \cdot u_{n,i}}{\sqrt{\sum_{i \in I_t} u_{t,i}^2} \cdot \sqrt{\sum_{i \in I_n} u_{n,i}^2}} \quad [13]$$

$$AC(u_t, u_n) = \frac{\sum_{i \in I_t \cap I_n} (u_{t,i} - \bar{u}_t) \cdot (u_{n,i} - \bar{u}_n)}{\sqrt{\sum_{i \in I_t \cap I_n} (u_{t,i} - \bar{u}_t)^2} \cdot \sqrt{\sum_{i \in I_t \cap I_n} (u_{n,i} - \bar{u}_n)^2}} \quad [14]$$

For the semantic similarity calculation, the t-LDA model was utilised to create the topic distribution matrix for all users. Specifically, each row of the topic-distribution matrix represented a user's topic-distribution vector θ_i, and the similarities between the target user and the other users were calculated via the semantic similarity method $Gensim.Similarities$ *Python library* refers to Equation 15 to select the top k neighbors and corresponding semantic similarity values. Nevertheless, the semantic similarity method was susceptible to the cold start and the richness of the contents.

$$\cos(u_t, u_n) = \frac{\sum_{t,n} \theta_t \cdot \theta_n}{\sqrt{\sum_t \theta_t^2} \cdot \sqrt{\sum_n \theta_n^2}} \quad [15]$$

For the traditional similarity calculation, the t-SENTIMENT model calculated the sentiment scores based on the posted tweets to generate the sentiment scores matrix for all users. Each row of the sentiment scores matrix represented a user's score vectors, and the similarities between the target user and the other users were calculated via the traditional similarity methods (cosine-based similarity, Pearson's correlation coefficient, and AC) to select the top k neighbors and obtain the similarity values. Nonetheless, traditional similarity methods utilised the score matrix to search for similar users, which did not account for semantic relationships between words. Therefore, a hybrid similarity method was proposed to address the limitations of the traditional similarity method and the semantic similarity method. The hybrid similarity method Equation 16 could assist in maximizing the accuracy and adaptability of the CF-RS. The hybrid ratio weight was ascertained through the pre-training process based on the evaluation metric, namely MAE.

$$Hsim_{t,n} = \lambda sim_{t,n}^{t-LDA} + (1-\lambda) sim_{t,n}^{t-Sentiment} \quad [16]$$

Sentiment Score Prediction by u-KNN

The K-Nearest Neighbor (KNN) is a technique for prediction based on K neighbors' data in CF, and CF can be subdivided into item-based and user-based CF. The user-based KNN (u-KNN) Equation 17 was employed in the current study to predict the sentiment scores of top-k topics on the generated recommendation matrix:

$$\hat{r}_{u,i}{}^{KNN} = \frac{\sum_{n \in N_{neig\ hbors}^{k}} sim_{u,n} \times r_{n,i}}{\sum_{n \in N_{neig\ hbors}^{k}} \left| sim_{u,n} \right|} \quad [17]$$

Where $N_{neig\ hbors}^{k}$ is the k neighbors who are similar to the target user *u*, $sim_{u,n}$ is the similarity between the target user *u* and neighbor *n*, $r_{n,i}$ is the score rated by neighbor *n* for item *i*.

RESULTS AND DISCUSSION

The current experiments provided the results of parameter optimisation for the employed algorithm, and the parameters included the number of topics, hyperparameters, neighborhood size, and hybrid ratio. In addition, the performance of the proposed algorithm was compared with the baseline algorithms under the optimal parameters.

Datasets

The proposed algorithm and baseline algorithms were applied to the dataset of tweets to compare the performance of the algorithms, which could aid in validating the effectiveness of the proposed algorithm, namely HDTS-uKNN. The tweet contents were extracted through the Python 3 library SNSCRAP. A query with keywords, such as (#Amazon OR Shopee), the specified time range, such as from "since:2022-07-13" to "until 2022-11-30", and the language set as “en” aided in collecting users’ interaction data. The data were collected over five months to ensure that a substantial amount of relevant information was gathered, and with an appropriate time span for analysis. Examples of the user collection data are depicted in Table 3.

Table 3
Example of the user collection data

Tweet	Content
Tweet1	Works deadly. Have one on my river boat. Best $200 Amazon buy ever
Tweet2	This looks like an interesting series, or it could be that I am the target audience. One problem with writing a book is that Amazon shows me ads for more books. Some days, I am spending close to what I am making from sales on buying books.
Tweet3	As a kid, I’d get embarrassed when my mother got riled up over the lack of acknowledgment of Chanukah (& other Jewish holidays) in school. They say everything is cyclical & here I am putting in an Amazon order of Chanukah decorations for my kid’s daycare room, so she feels included.

The preprocessing steps were performed rigorously, which included (1) removing duplicate tweets based on username and content, (2) converting emoji to text description by utilising the emoji of Python, (3) lowercasing texts and removing links and special characters from texts through a regular expression from the built-in library of Python, (4) tokenising the data by breaking down the collected data into smaller units or tokens (individual words), (5) lemmatising words by applying the WordNet lemmatiser from the Natural Language Toolkit (NLTK) of Python. Nevertheless, removing specific words, such as “no”, “not”, and “will”, would impact the accuracy of the sentiment analysis model. Thus, the stop words were retained to avoid the impact on subsequent results. A total of 400,000 tweets were collected, and the dataset was divided into 80% for training and 20% for testing to predict the target user’s sentiment score. Particularly, the proposed algorithm, namely HDTS, was applied to generate the user-topic distribution matrix and user-topic sentiment score matrix on the processed content. Subsequently, the user-topic distribution matrix and user-topic sentiment score matrix were divided according to the username to evaluate the performance of the proposed methods. To obtain the optimal parameters, the training data were also divided into 20% as validation-test data and 80% validation-training data for tuning, such as topics K, hyperparameters α,β, neighborhood size, and hybrid ratio λ.

Evaluation Metrics

Two types of error evaluation metrics were employed, namely the mean absolute error (MAE) and root mean square error (RMSE). The MAE was utilised to compute the absolute error between predicted sentiment scores and actual sentiment scores, as defined in Equation 18. The RMSE was applied to calculate the square root of the absolute error between predicted sentiment scores and actual sentiment scores, as defined similarly to the MAE in Equation 19. The current study concentrated on the MAE as the primary metric due to the MAE providing a more balanced picture of general model performance across all predictions.

$$MAE = \frac{\sum_{u,i \in T} |S_{ui} - S'_{ui}|}{|T|} \quad [18]$$

$$RMSE = \sqrt{\frac{\sum_{u,i \in T} \left|S_{ui} - S'_{ui}\right|^2}{|T|}} \quad [19]$$

To determine the values of optimal hyperparameters α, β, and the number of topics K for the topic model (LDA), the coherence method was employed for the evaluation, with the formula denoted as Equation 20:

$$Coherence(D_{text}) = \frac{2}{N(N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} PMI(W_i \cdot W_j)$$

$$PMI(W_i, W_j) = \log\left(\frac{P(W_i, W_j)}{P(W_i) \cdot P(W_j)}\right) \quad [20]$$

Where PMI is a measure of the degree of association between words, $P(W_i)$ and $P(W_j)$ denote the probability that the i_{th} and j_{th} words occur individually in the specific topic, $P(W_i, W_j)$ denotes the probability that the i_{th} and j_{th} words occur concurrently in the specific topic, and N denotes the number of words in the topic.

Preliminary Experiment for Parameter Optimisation

The training data were divided into 80% as validation-training data and 20% as validation-test data for tuning the optimal parameters to determine the optimal parameters, including the number of topics K, hyperparameters α, β, neighborhood size, and hybrid ratio λ. Additionally, another parameter, namely *optimise_interval*, was included to optimise the hyperparameters, as the topic model was LDA via Mallet, and the β was a fixed value that could not be stipulated directly. In the parameter-turning process, the control variable method was utilised to adjust the parameters sequentially, and other related parameters remained unchanged. In the proposed topic model (t-LDA), four parameters, namely the number of topics K, hyperparameters α, β, and *optimise_interval*, were optimised.

Specifically, topics K represents the number of topics in the topic model, α symbolises the topic distribution of each document, and β emblematises the word distribution of each topic. The hyperparameter α and *optimise_interval* were set to default values, the number of topics K ranged from 10 to 200, and the number of topics increased by 10 each time to ascertain the optimal coherence value of the topic model (t-LDA). Figure 3 illustrates that the coherence value reaches its highest when the number of topics K is 90. Figure 4 depicts that the coherence value achieves the optimal value when the alpha is 1 and topic K is 90.

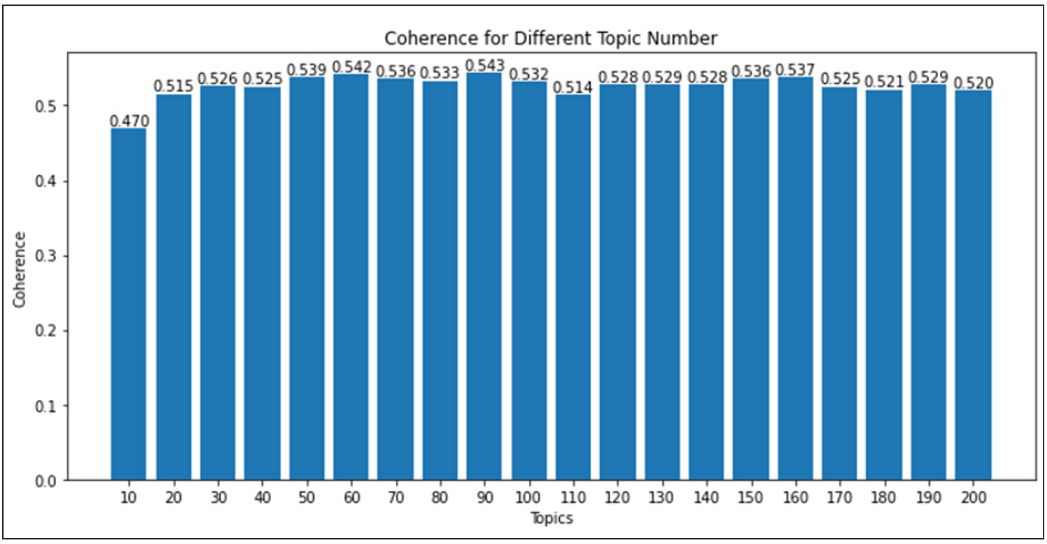

Figure 3. The coherence values for different topic numbers

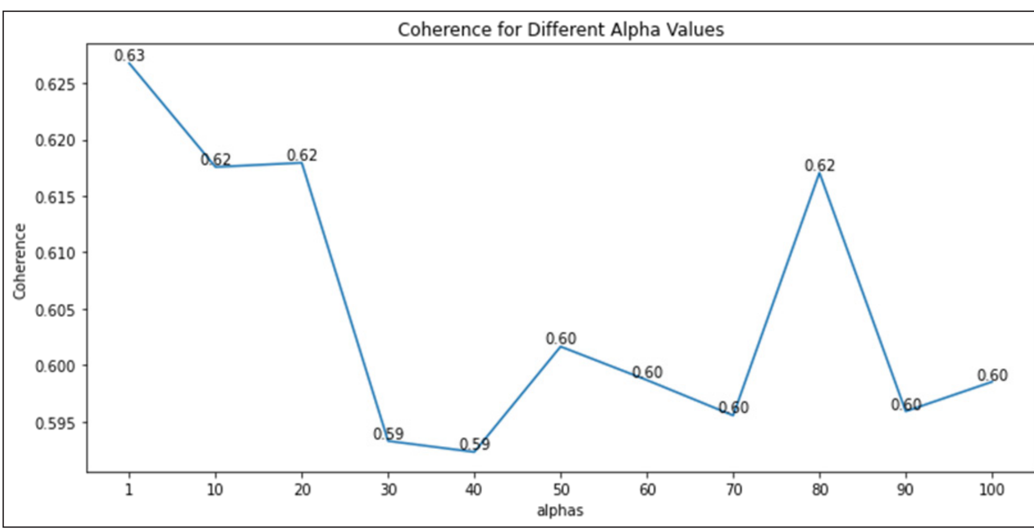

Figure 4. Coherence values for different alpha values

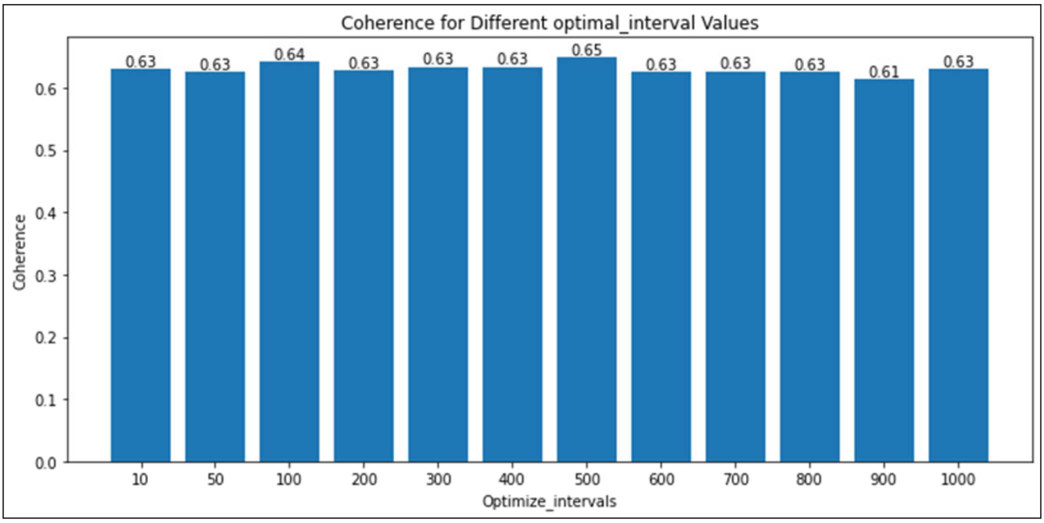

Figure 5. Coherence for different optimal_interval values

The topics *K* and alpha values were set to 90 and 1, the *optimal_interval* ranged from 10 to 1000, and the coherence values are portrayed in Figure 5. The coherence value was close to the optimal value when *optimal_interval* was 500. Therefore, the topics *K*, alpha, and *optimal_interval* values were set to 90, 1, and 500, respectively, and the parameter values were further chosen as the optimum parameters of the t-LDA for the following experiments.

In the proposed hybrid similarity method, two parameters, namely the neighborhood size and hybrid ratio, were optimised. The optimal traditional similarity method was also selected for hybrid similarity calculation. The performance of traditional similarity methods was compared based on the same neighborhood-based prediction method (u-KNN). Subsequently, the most optimal traditional similarity method was selected to be combined with the semantic similarity method as the hybrid similarity method to address the limitation of the traditional similarity method in terms of sparse data. The MAE values varied with the parameter change, which were calculated based on the prediction sentiment scores and the true sentiment scores for all target users. The true sentiment scores of the target user were generated through the t-SENTIMENT model, and the prediction sentiment scores of the target user were calculated (u-KNN) based on the sentiment scores of top K neighbors and hybrid similarity values. All users from the validate-test data were treated as the target users, the top k neighbors were obtained from the validate-training data based on the hybrid similarity values, and the MAE values of target users about all topics were computed for each neighbor number. Figure 6 shows that the cosine similarity method accomplishes the lowest prediction error compared to the other two traditional similarity methods, namely Pearson’s correlation coefficient and AC, at the same neighborhood size. The prediction

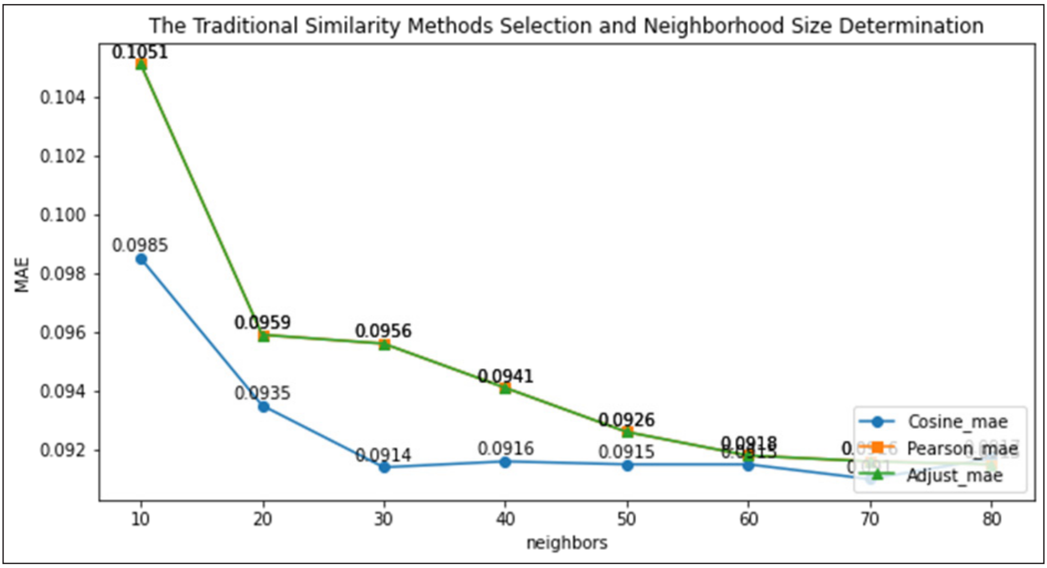

Figure 6. Comparison between the traditional similarity methods

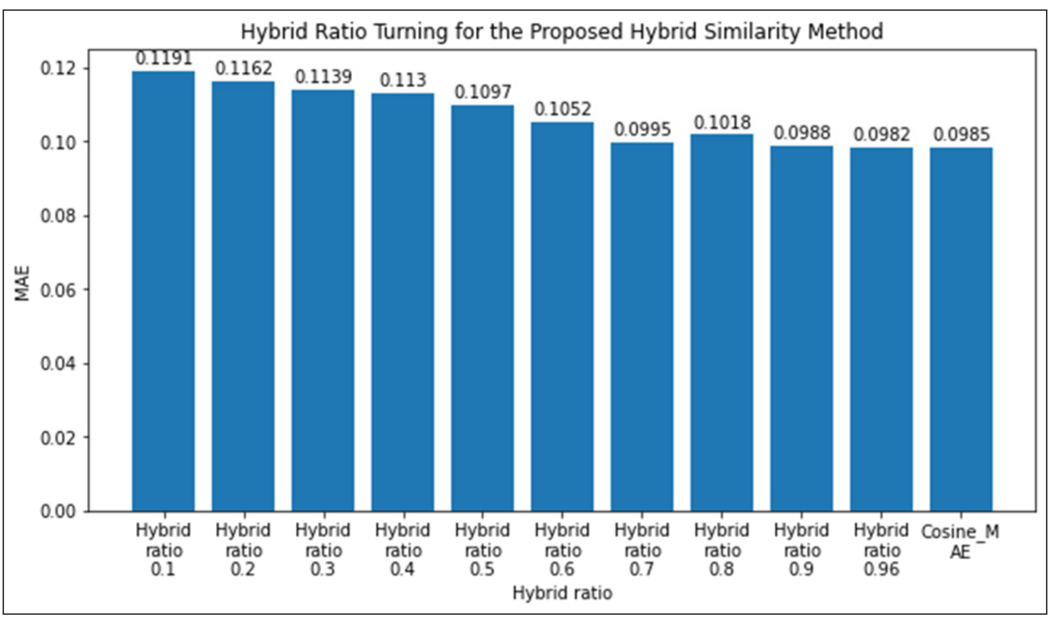

Figure 7. Hybrid ratio adjustment

error became smaller when the neighborhood size increased. Notwithstanding, all three traditional similarity methods demonstrated the most serious data sparsity issue when the neighborhood size was 10. Compared to the traditional similarity methods that were constantly susceptible to the influence of the number of scores, the semantic similarity method was not impacted by the sufficiency or sparseness of scores during the similarity computation process. Hence, the semantic similarity method was employed to enhance the prediction performance of the traditional similarity method (cosine similarity) when the neighborhood size was 10.

Figure 7 illustrates the hybrid ratio of the semantic similarity method combined with the traditional similarity method (cosine similarity). The hybrid similarity method exhibited the lowest prediction error of 0.0982 compared to the prediction error of 0.0985 of cosine similarity when the hybrid ratio reached 0.96 on the validation-test data. The hybrid similarity method considered both the advantages of users' personal sentiment scores and the latent users' preferences on the tweets dataset, which aided in improving the interpretability and effectiveness of the proposed RS.

Sentiment Score Prediction Results

The effectiveness of the hybrid similarity method and the time decay function was compared on the test data to predict the sentiment score of the retail customers. The MAE values were computed based on the prediction sentiment scores and the true sentiment scores of all target users about all topics. Specifically, the t-SENTIMENT model was applied to generate the true sentiment scores of the target user, and the predicted sentiment scores of the target user were calculated (u-KNN) based on the sentiment scores of top K neighbors and hybrid similarity values. Each user from the test data was treated as the target user, and the top k neighbors were obtained from the training data based on the hybrid similarity values. Moreover, the proposed hybrid similarity method and baseline methods were compared on the output of the proposed model t-SENTIMENT, with the predicted results demonstrated in Figure 8. The proposed hybrid similarity method exhibited the lowest prediction error of 0.1003 compared to the baseline similarity methods (0.1007 for the cosine similarity method, 0.1071 for the Pearson similarity method, 0.1071 for the AC similarity method, and 0.1380 for the semantic similarity method). The results suggested that the proposed similarity method assisted in enhancing the understanding of personal sentiment variation and latent user preference, which could further improve the predictive capability of the CF algorithm. Compared to the baseline similarity calculation methods that employed the score matrix to calculate similarity, the proposed hybrid similarity method not only could more accurately capture the relationship among the semantics of users' contents but also could account for users' sentiments. The hybrid similarity calculation utilised multiple features (sentiment scores and topic distribution) to compute the similarity between target users and neighbors, which could prevent similarity errors caused by single features due to noise, bias, sparsity, and cold start.

The MAEs of traditional similarity methods were lower compared to the proposed semantic similarity method (0.1380), as the semantic similarity method calculated

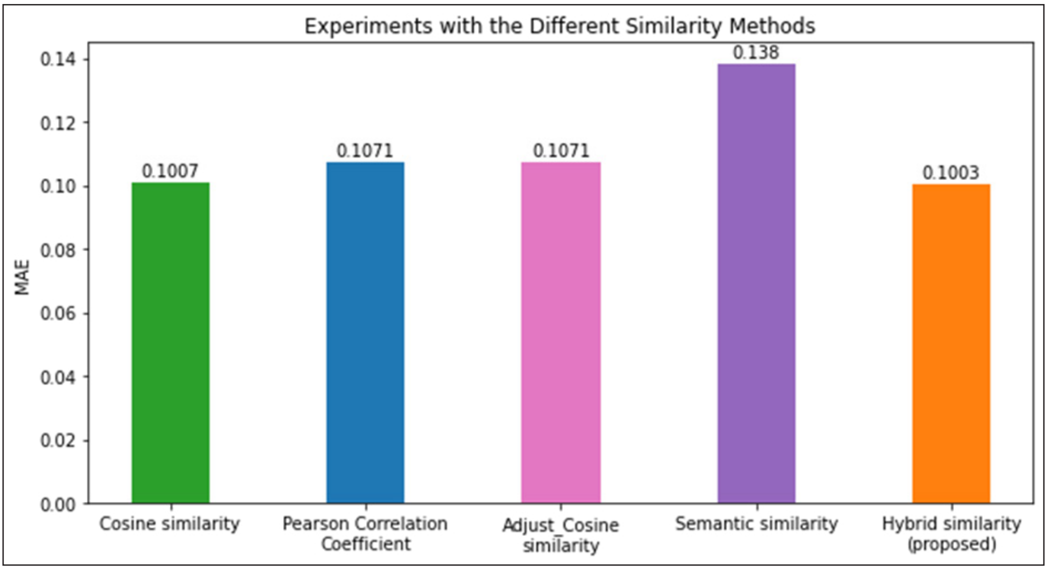

Figure 8. Performance results of different similarity methods

similarity based on topic features, which allowed for capturing latent user preferences more effectively. Hence, the semantic similarity method was particularly suitable for sparse data situations, such as ratings. Meanwhile, the semantic similarity method contained certain limitations, as the method could be influenced by cold start and content richness. The method could not accurately compute similarity values between users based on topic features when users did not possess relevant topic descriptions (cold start) or the content was insufficiently rich. In addition, the characteristics of the three different traditional similarity methods contributed to the differences. Specifically, the cosine similarity method calculated the similarity between users by considering the sentiment scores vector, whereas the Pearson's correlation coefficient (PCC) method computed the similarity between users by accounting for the relative deviation of the user's sentiment scores and the user's co-rated average score. Thus, PCC could aid in preventing inaccurate similarity calculation engendered by the users' score habits. Comparatively, the AC similarity method calculated the similarity between users through the relative deviation of the users' sentiment scores and the user's average score, which demonstrated a highly similar definition to the PCC in the user-based condition. The primary limitations of the PCC and AC were that the target user's score pattern differed significantly from the score patterns of other users, and the data sparsity issue led to inaccurate similarity calculations between the score patterns of the target user and the top k neighbors. The issue caused variations in score patterns among users, resulting in higher prediction errors compared to cosine similarity.

The proposed hybrid similarity method minimally reduced the MAE from 0.1007 (Cosine) to 0.1003 (HDTS-uKNN), as depicted in Figure 8. The reason for the minor

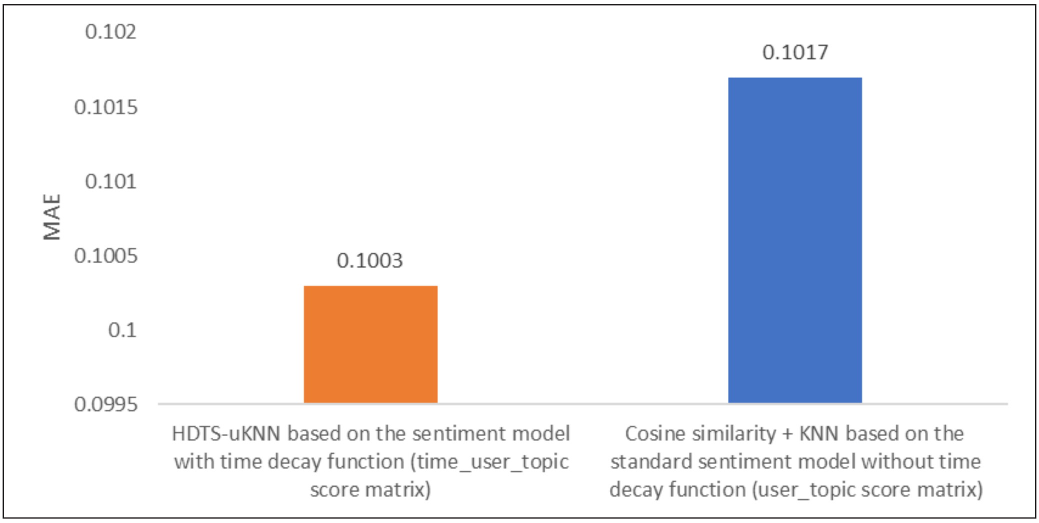

Figure 9. The comparison between HDTS-uKNN based on the t-SENTIMENT and cosine similarity based on the standard sentiment model

difference was that all similarity methods (hybrid similarity, cosine similarity, Pearson similarity, AC similarity, and semantic similarity) were based on the output results (time-user_topic score matrix) of the proposed model (t-SENTIMENT). The time decay function reduced the influence of outdated scores and noise on the similarity calculation process. Therefore, the predictive effect (MAE) of the proposed hybrid similarity method was not profound compared to cosine similarity. Figure 9 illustrates the comparison between the proposed method (HDTS-uKNN) based on the output results (time_user_topic score matrix) of the proposed model (t-SENTIMENT) and the baseline method (cosine similarity) based on the output results (user_topic score matrix) of the standard sentiment model (VADER). The proposed hybrid similarity method (0.1003) achieved a valid improvement in predictive MAE compared to the baseline method (0.1017). From the e-commerce perspective, a reduction of 0.0014 in the MAE could change the top k recommendations, which would translate into considerable revenue growth and increasing sales. From the theoretical perspective, the proposed method could be complementary to deep learning and transformer models to enhance interpretability and lower computational resource consumption.

Effect of Time Decay on the Prediction

The second experiment was conducted by utilising the same dataset to corroborate the effect of the time decay on the prediction by including the time decay function in the sentiment model called (t-SENTIMENT). Particularly, the user_topic sentiment matrix with and without time was applied to compare the prediction performance of the hybrid similarity method and the baseline method to ascertain the superiority of the proposed

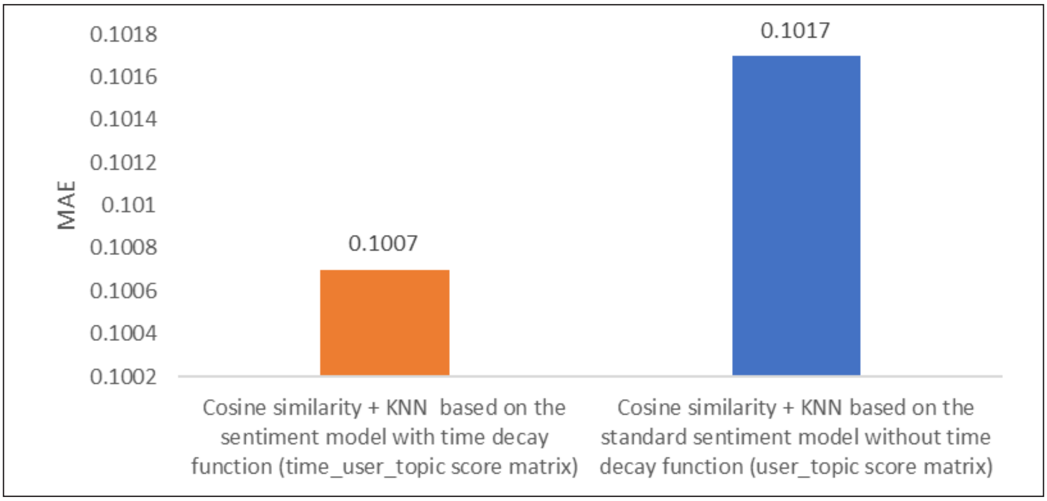

Figure 10. Impact of the time decay function on the sentiment model for cosine similarity-KNN prediction

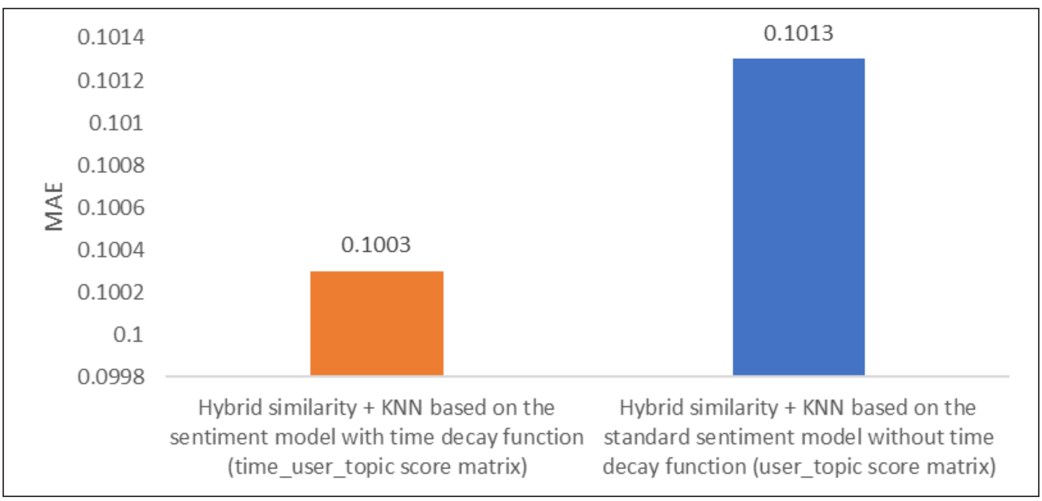

Figure 11. Impact of the time decay function on the sentiment model for hybrid similarity-KNN prediction

t-SENTIMENT model. The separate experimental results of the cosine similarity and hybrid similarity with KNN based on the user_topic sentiment matrix with and without time are illustrated in Figures 10 and 11.

Figure 10 demonstrates that the cosine similarity method with KNN produces a lower prediction error (0.1007) based on the time_user_topic sentiment matrix generated by the t-SENTIMENT model, compared to the cosine similarity method with KNN on the matrix generated by the standard sentiment model (0.1017). Figure 11 depicts the comparison result, which also demonstrates that the hybrid similarity method with KNN based on the sentiment model with the time decay function achieves a lower prediction error (0.1003), compared to the hybrid similarity method with KNN based on the standard sentiment model without the time decay function (0.1013). The difference was owing to the proposed t-SENTIMENT model that assigned different time weights to the sentiment scores based on the time factor, which reduced the influence of outdated scores or noise on the similarity calculation process to minimise the prediction error. The time decay factor not only aided CF in managing user preference changes but also determined different time weights, which enabled the proposed algorithm to focuss more on the user's recent scores.

The experimental results validated the effectiveness of the proposed method, namely HDTS-uKNN, on Twitter data. Despite the data sparsity, semantic ignorance, and temporal shifts being generic and pervasive challenges in RSs, the issues are more significant on Twitter. Hence, the proposed method was discovered to be more suitable compared to baseline methods for predicting user preferences from real social media data. While the dataset was Twitter-based, and search queries on Amazon and Shopee might not fully represent true consumer preferences, the Twitter data could serve as supplementary information to bridge the gaps in e-commerce datasets to assist in improving recommendation accuracy. Summarily, the current study contributed a lightweight and interpretable recommendation strategy for resource-constrained merchants by exploring the adaptability of CF combined with dynamic, sentiment, topic, and hybrid similarity approaches on social media data.

CONCLUSION AND FUTURE DIRECTIONS

An enhanced sentiment score prediction in the CF algorithm was proposed in the present study, which employed the hybrid similarity calculation and u-KNN to assist retailers in more effectively leveraging social media data from e-commerce platforms to maximise revenue and increase the ability of retailers to interact with users. The proposed hybrid similarity calculation method incorporated two enhanced models, namely the semantic model t-LDA and the sentiment model t-SENTIMENT, which were applied to understand the semantic, latent user preferences, and users' opinions from users' tweets for items. The first experiment results demonstrated that the proposed hybrid similarity calculation method was superior to other baseline methods, and the hybrid similarity method improved the adaptability of the CF algorithm, especially when the data encountered issues, such as sparsity and noise. The second experiment results revealed that VADER produced more reliable scores by integrating the time decay function, which produced more reliable recommended items. Time decay integration not only could aid CF in focussing more on the user's recent scores but also assist in minimising the influence of outdated scores or noise on the calculation process. Therefore, the current study validated the performance of CF, which was combined with LDA, VADER, the time decay function, and the hybrid similarity method, on the large-scale Twitter dataset. Meanwhile, certain limitations exist. For instance, the hyperparameter α optimisation was conducted in the range of 1 to 100, which identified the optimal value at $\alpha = 1$. The lower range of α (e.g., [0.01, 1]) was not explored in the present study. Hence, future researchers can explore a different range of $\alpha \in [0.01, 1.0]$ to assess the influence on the coherence of the proposed model. Moreover, more robust data embedding techniques and semantic models, such as ELMo and BERT, can be applied to account for the order of words, the dimensionality of the feature space, scalability, and context awareness.

ACKNOWLEDGEMENT

The authors sincerely appreciate the valuable support from the School of Computer Sciences, Universiti Sains Malaysia. This work was supported by the [Ministry of Higher Education Malaysia] (Grant numbers [FRGS/1/2020/ICT06/USM/02/3]).

REFERENCES

Abbasi-Moud, Z., Vahdat-Nejad, H., & Sadri, J. (2021). Tourism recommendation system based on semantic clustering and sentiment analysis. *Expert Systems with Applications, 167*, Article 114324. https://doi.org/10.1016/j.eswa.2020.114324

Asani, E., Vahdat-Nejad, H., & Sadri, J. (2021). Restaurant recommender system based on sentiment analysis. *Machine Learning with Applications, 6*, Article 100114. https://doi.org/10.1016/j.mlwa.2021.100114

Chatti, M. A., Guesmi, M., & Muslim, A. (2024). Visualisation for recommendation explainability: A survey and new perspectives. *ACM Transactions on Interactive Intelligent Systems, 14*(3), 1-39. https://doi.org/10.1145/3672276

Chehal, D., Gupta, P., & Gulati, P. (2021). Implementation and comparison of topic modelling techniques based on user reviews in e-commerce recommendations. *Journal of Ambient Intelligence and Humanized Computing, 12*(5), 5055-5070. https://doi.org/10.1007/s12652-020-01956-6

Chen, H., Chen, H., Wang, Y., & Li, J. (2025). Research on optimisation of personalised algorithm of e-commerce recommendation system based on deep learning. In *Proceedings of the 2025 International Conference on Digital Analysis and Processing, Intelligent Computation (DAPIC 2025)* (pp. 591-595). IEEE. https://doi.org/10.1109/DAPIC66097.2025.00115

Chen, J., Dong, H., Wang, X., Feng, F., Wang, M., & He, X. (2023). Bias and debias in recommender systems: A survey and future directions. *ACM Transactions on Information Systems, 41*(3), 1-38. https://doi.org/10.1145/3564284

Chen, Y. C., Hui, L., & Thaipisutikul, T. (2021). A collaborative filtering recommendation system with dynamic time decay. *The Journal of Supercomputing, 77*(1), 244-262. https://doi.org/10.1007/s11227-020-03266-2

Devika, P., & Milton, A. (2025). Book recommendation using sentiment analysis and ensembling hybrid deep learning models. *Knowledge and Information Systems, 67*(2), 1131-1168. https://doi.org/10.1007/s10115-024-02250-z

Garg, S. (2021). Drug recommendation system based on sentiment analysis of drug reviews using machine learning. In *Proceedings of the 11th International Conference on Cloud Computing, Data Science and Engineering (Confluence 2021)* (pp. 175-181). IEEE. https://doi.org/10.1109/Confluence51648.2021.9377188

Gong, S. J., Wu, Y. H., & Tan, H. S. (2009). Combining memory-based and model-based collaborative filtering in recommender systems. In *Proceedings of the 2009 Pacific-Asia Conference on Circuits, Communications and Systems (PACCS 2009)* (pp. 690-693). IEEE. https://doi.org/10.1109/PACCS.2009.66

Jain, G., Mahara, T., Sharma, S. C., Agarwal, S., & Kim, H. (2022). TD-DNN: A time decay-based deep neural network for recommendation systems. *Applied Sciences, 12*(13), Article 6398. https://doi.org/10.3390/app12136398

Jangid, M., & Kumar, R. (2025). Deep learning approaches to address cold start and long-tail challenges in recommendation systems: A systematic review. *Multimedia Tools and Applications, 84*(5). https://doi.org/10.1007/s11042-024-20262-3

Jelodar, H., Wang, Y., Rabbani, M., & Ayobi, S. (2019). Natural language processing via LDA topic model in recommendation systems [Preprint]. *arXiv*.

Jelodar, H., Wang, Y., Yuan, C., Feng, X., Jiang, X., Li, Y., & Zhao, L. (2019). Latent dirichlet allocation (LDA) and topic modelling: Models, applications, a survey. *Multimedia Tools and Applications, 78*(11). https://doi.org/10.1007/s11042-018-6894-4

Jiang, X., Bai, L., Yan, X., & Wang, Y. (2023). LDA-based online intelligent courses recommendation system. *Evolutionary Intelligence, 16*(5), 1619-1625. https://doi.org/10.1007/s12065-022-00810-2

Kaveri, V. V., & Maheswari, V. (2019). A framework for recommending health-related topics based on topic modelling in conversational data (Twitter). *Cluster Computing, 22*, 10963-10968. https://doi.org/10.1007/s10586-017-1263-z

Kumar, S., De, K., & Roy, P. P. (2020). Movie recommendation system using sentiment analysis from microblogging data. *IEEE Transactions on Computational Social Systems, 7*(4), 915-923. https://doi.org/10.1109/TCSS.2020.2993585

Liu, N., & Zhao, J. (2023). Recommendation system based on deep sentiment analysis and matrix factorisation. *IEEE Access, 11*, 16994-17001. https://doi.org/10.1109/ACCESS.2023.3246060

Na, L., Ming-Xia, L., Hai-Yang, Q., & Hao-Long, S. (2021). A hybrid user-based collaborative filtering algorithm with topic model. *Applied Intelligence, 51*(11), 7946-7959. https://doi.org/10.1007/s10489-021-02207-7

Rafailidis, D., & Nanopoulos, A. (2016). Modelling users' preference dynamics and side information in recommender systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 46*(6), 782-792. https://doi.org/10.1109/TSMC.2015.2460691

Rizkallah, S., Atiya, A. F., & Shaheen, S. (2021). New vector-space embeddings for recommender systems. *Applied Sciences, 11*(14), Article 6477. https://doi.org/10.3390/app11146477

Song, D., Li, Z., Jiang, M., Qin, L., & Liao, L. (2019). A novel temporal and topic-aware recommender model. *World Wide Web, 22*(5), 2105-2127. https://doi.org/10.1007/s11280-018-0595-9

Sun, L., Guo, J., & Zhu, Y. (2019). Applying uncertainty theory into the restaurant recommender system based on sentiment analysis of online Chinese reviews. *World Wide Web, 22*(1), 83-100. https://doi.org/10.1007/s11280-018-0533-x

Tahmasbi, H., Jalali, M., & Shakeri, H. (2021). Modelling user preference dynamics with coupled tensor factorisation for social media recommendation. *Journal of Ambient Intelligence and Humanized Computing, 12*(10), 9693-9712. https://doi.org/10.1007/s12652-020-02714-4

Talhaoui, M. A., El Bour, H. A., Moulouki, R., Nkiri, S., & Azouazi, M. (2018). An improved social media analysis on three layers: A real-time enhanced recommendation system. *International Journal of Advanced Computer Science and Applications, 9*(2), 242-247. https://doi.org/10.14569/IJACSA.2018.090234

Tang, Z., Zhang, X., & Niu, J. (2020). LDA model and network embedding-based collaborative filtering recommendation. In *Proceedings of the 6th International Conference on Dependable Systems and Their Applications (DSA 2019)* (pp. 283-289). IEEE. https://doi.org/10.1109/DSA.2019.00043

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems, 30*.

Wedel, M., & Kannan, P. K. (2016). Marketing analytics for data-rich environments. *Journal of Marketing, 80*(6), 97-121. https://doi.org/10.1509/jm.15.0413

Wilson, J., Chaudhury, S., & Lall, B. (2014). Improving collaborative filtering-based recommenders using topic modelling. In *Proceedings of the 2014 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology Workshops (WI-IAT 2014)* (pp. 340-346). IEEE. https://doi.org/10.1109/WI-IAT.2014.54

Yuwono, T., Suroso, A., & Novandari, W. (2024). Information and communication technology in SMEs: A systematic literature review. *Journal of Innovation and Entrepreneurship, 13*(1), Article 92. https://doi.org/10.1186/s13731-024-00392-6

Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). Deep learning-based recommender systems: A survey and new perspectives. *ACM Computing Surveys, 52*(1), 1-35. https://doi.org/10.1145/3285029

Zhang, Y., & Chen, X. (2020). Explainable recommendation: A survey and new perspectives. *Foundations and Trends in Information Retrieval, 14*(1), 1-101. https://doi.org/10.1561/1500000066

Zhou, J., Jiang, G., Du, W., & Han, C. (2023). Profiling temporal learning interests with time-aware transformers and knowledge graph for online course recommendation. *Electronic Commerce Research, 23*(4), 2357-2377. https://doi.org/10.1007/s10660-022-09541-z

Zhu, K., & Samsudin, N. H. (2024). Attention-based spatialised word embedding Bi-LSTM model for sentiment analysis. *Pertanika Journal of Science and Technology, 32*(1), 79-98. https://doi.org/10.47836/pjst.32.1.05