ORIGINAL ARTICLE

A sequential neural recommendation system exploiting BERT and LSTM on social media posts

A. Noorian[1] · A. Harounabadi[1] · M. Hazratifard[2]

Abstract

Tourists share opinions about Points of Interest (POIs) through online posts and social media platforms. Opinion mining is a popular technique for extracting feedback from tourists who visited various places hidden in reviews, which are used in several tourist applications that generally reflect their preference towards POI. On the other hand, a trip schema is difficult for tourists because they must pick up sequential POIs in unknown areas that meet their limitations and preferences. However, most prior trip suggestion methods are suboptimal for several reasons, including that they do not consider valuable user reviews and rely exclusively on left-to-right unidirectional discovery sequence models. This study proposes a Neural Network-Long Short-Term Memory (LSTM) POI recommendation system for calculating user similarity based on opinions and preferences. In addition, it presents a method for discovering sequential trip recommendations with Bidirectional Encoder Representations from Transformer (BERT) using a deep learning method. Furthermore, this neural hybrid framework identifies a list of optimal trip candidates by combining personalized POIs with multifaceted context. Furthermore, this method employs the valuable information contained in user posts and their demographic information on social media to mitigate the well-known cold start issue. In the experimental evaluation based on two datasets, Tripadvisor and Yelp, this hybrid method outperforms other state-of-the-art methods when considering F-Score, nDCG, RMSE, and MAP.

Keywords Natural language processing · Transformer · POI · Social media · LSTM · Deep learning

Introduction

The exponential increase in the amount of information and the number of Internet users has created the potential problem of information overload, which hinders the search for relevant items on the Internet. Recommender systems (RSs) assist users in receiving personalized suggestions, thereby assisting them in making the best decisions when conducting online transactions [1]. As the field of recommender systems

has progressed, some researchers have employed machine-learning techniques from the field of artificial intelligence to generate more accurate recommendations. Selecting an appropriate machine-learning algorithm for a recommender system is challenging due to the abundance of such algorithms.

A recommendation system is defined as a type of machine learning that utilizes data to predict, narrow down, and locate what individuals seek among an exponentially increasing number of options. In the past, recommender systems have relied on clustering, nearest neighbor, and matrix factorization techniques. However, deep learning has achieved tremendous success in recent years across multiple domains, including image recognition and natural language processing (NLP). The success of deep learning has benefited recommender systems as well. Modern recommender systems, such as those at YouTube and Amazon, are powered by complex deep learning systems and less by conventional methods [2]. Applying deep learning to a recommendation system that must mine and extract features from vast amounts of data will not only aid in the development of recommendation

✉ A. Harounabadi
  A.Harounabadi@iauctb.ac.ir

  A. Noorian
  Ali.Noorianavval.Eng@iauctb.ac.ir

  M. Hazratifard
  Hazrati@cse.shirazu.ac.ir

1 Department of Computer Engineering, Central Tehran Branch, Islamic Azad University, Tehran, Iran

2 Department of Computer Science and Engineering, Shiraz University, Shiraz, Iran

algorithms but will also improve algorithm performance and, consequently, the user experience [3].

Conventional collaborative filtering (CF) methods provide recommendations based on the travel preferences of users who behave similarly to the targeted tourists. In real-world applications, however, user similarity may vary, indicating that most contemporary symmetric techniques produce less accurate results [4]. Context-aware (CA) recommender systems, on the other hand, consider the user's context and provide more accurate suggestion outputs, given that numerous tools now collect data on the user's state [5].

The cold start problem is a significant issue with recommender systems in which RSs may not be able to recommend suitable items because of insufficient information about new users. In addition, the sparsity issue occurs in some RSs when the number of user rankings is significantly fewer than the number of things; consequently, the RS is weak in predicting significant evaluations, and conventional techniques may result in unsuitable suggestions [6]. On the other hand, the increasing popularity of social and e-commerce media sites has encouraged users to write reviews that naturally describe their evaluations of products. These comments typically take the form of textual explanations of why the reviewer likes or dislikes a product based on their personal experiences. The system can capture the multifaceted nature of a user's opinions from their reviews and, as a result, construct a preference model for the user that cannot be obtained from overall ratings. Various review-based recommender systems have been developed in recent years to incorporate this information from user-generated textual reviews into user modeling and to recommend the process. Furthermore, utilizing the valuable information in social media user posts and reviews has proven to be particularly effective in addressing rating sparsity and cold start issues [7, 8].

A tourism Recommendation System is a specific recommender system for the tourism industry that provides valuable suggestions and guidance to tourists in locating Points of Interest (POIs), identifying amenities such as transport, hotel, and attractions according to their preferences, interests, likes, and budget to make their trip a memorable one [9, 10]. In recent years, there has been an increase in the number of articles utilizing Location-Based Social Networks (LBSN) and spatial–temporal contextual data in tourist RS. On the other hand, most modern recommender systems have some drawbacks, primarily due to the dynamic nature of tourists' travel habits, which makes designing recommender systems for tourism purposes difficult and complex.

Because selecting and organizing POIs can be difficult for travelers unfamiliar with the variety of locations in a new area, the planning process for a custom trip can be time-consuming [11]. In other words, visitors prefer pre-arranged POIs to a list of suggested POIs; therefore, a tourist will find an RS that can generate pre-arranged POIs more advantageous [12].

In this field, numerous techniques for modeling sequential recommendations for journey recommendations have been proposed. Most sequential recommendation models are based on deep learning and past user interactions [13]. However, applying NLP techniques to POI recommender systems is currently intriguing and is regarded as one of the most innovative ideas in sequential recommendations. NLP techniques enable machines to interpret and comprehend human language. In novel ways, transformers and large language models such as bidirectional-encoder-representations-from-transformers (BERT) have been applied to recommendations.

BERT is a novel NLP-based model developed at the Google Research Center [14]. This algorithm has resolved several NLP problems with high accuracy [15]. These language models are designed to learn the semantics of sentences by modeling which words match which other words and a similar strategy can be applied to recommend items by examining sequences of items rather than sentences of words.

Due to the complexity of recommender systems, hybrid techniques may be used to improve performance, particularly with the rise of social networking. Several hybrid techniques have recently been applied to recommender systems [16, 17]. This study proposes a novel neural hybrid technique for enhancing model performance and overcoming the drawbacks mentioned above. Our framework applies a combination method to CF based on deep learning, demographic-based (DB), and BERT to overcome the limitations of prior techniques. This work provides acceptable tourist destinations and itineraries based on evolving tourist preferences. The study also addresses the issue of cold starts by utilizing demographic data and user feedback. Asymmetric schema resolves problems associated with symmetric user similarity and improves algorithm performance.

Nowadays, social media platforms generate an enormous amount of data every day, making it difficult for users to find and discover relevant content. As a result, there is a need for efficient recommendation systems that can provide personalized and relevant content to users based on their preferences and behavior. Existing recommendation systems rely on traditional CF or content-based approaches, which have limitations in capturing the complex relationships between users and items.

The main aspect of our novelty and advance in the described idea is the use of BERT and Long Short-Term Memory (LSTM) to capture both the semantic and sequential information in social media posts. By exploiting the sequential representations of BERT and the contextualized dependencies of LSTM, the proposed model can provide more accurate and relevant recommendations to users. This is a significant improvement over existing recommendation

systems that do not take into account social media posts. The proposed model can handle a variety of recommendation tasks, including personalized content recommendation, social influence modeling, and social network analysis. Furthermore, the model can be applied to a wide range of social media platforms, such as Flicker, Facebook, and Instagram.

Concerning other ideas in the field, the proposed model is different from traditional RSs approaches in that it leverages the power of deep learning to capture both the sequential and semantic information in social media posts. The model is also different from other deep learning-based recommendation models that use only one type of neural network, such as a convolutional neural network (CNN) or a recurrent neural network (RNN).

With the increasing volume of social media data and the need for more accurate and personalized recommendations, the research topic of tourism recommendation systems based on sentiment analysis has emerged as a very new and promising area of study [18]. The great potential of such systems in the tourism industry has attracted researchers' attention. In this paper, we propose a novel approach that combines the power of BERT and LSTM models to enhance the quality of recommendations in the tourism domain. Unlike existing approaches that solely rely on ranking, our method utilizes user reviews to consider both the tourists' preferences and the sentiment associated with various tourist attractions. By leveraging the contextualized representations of LSTM and the semantic understanding of BERT, our model can provide more accurate and personalized recommendations tailored to individual tourists' preferences. Additionally, our approach demonstrates versatility and scalability, making it applicable to various tourism platforms and recommendation tasks. Through these advancements, our proposed approach aims to significantly improve the recommendation performance in the tourism industry and contribute to the field of recommender systems.

Our method receives trip suggestion requests from new users in a city. It utilizes geotagged data and user reviews as input data, which are then processed in our model. The model generates recommendations for the best trips, including top-N POIs, as output. Our approach takes into account contextual and demographic information, sequential historical POIs, and semantic user reviews throughout this process.

This study serves as an essential supplement to our previous paper [19]. This new method enhanced the previous methodology by incorporating additional important tourist factors. Additionally, our new framework uses user reviews to improve collaborative filtering recommendations. Some travelers prefer to provide an exhaustive review of their trip. These descriptions, accessible as user reviews, can assist recommender systems in locating tourists with comparable preferences. We propose a model that employs LSTM, an artificial intelligence technique, to enhance the collaborative filtering model and improve the estimation of user similarity.

This approach proposes a new neural Network-LSTM and BERT-based POI recommendation system that considers valuable user reviews and preferences to generate personalized and accurate trip recommendations. Unlike most prior trip suggestion methods that rely on left-to-right unidirectional discovery sequence models, this method utilizes deep learning models to generate accurate and relevant recommendations based on user feedback and behavior. The proposed method combines personalized POIs with multifaceted contexts, such as user demographics and location, to identify a list of optimal trip candidates. Additionally, this method addresses the cold start issue by incorporating user posts and demographic information from social media. The experimental evaluation of this method on two datasets, Tripadvisor and Yelp, outperforms other state-of-the-art methods, demonstrating its effectiveness in generating personalized and accurate trip recommendations.

The main contributions of this research are outlined as follows:

- Developing an LSTM model for discovering users' similarities based on posts on social media
- Proposing a personalized POI trip framework using contextual, demographic data and user opinions
- Applying BERT for locating Sequential POIs

This research is organized as follows: "Background knowledge" reviews the background knowledge of our research. The problem formulation and our proposed method, called BERT-LSTM-hybrid, are presented in "The proposed method". "Simulations and experimental evaluation" further discusses the proposed methodology and its testing and evaluation results compared with the existing papers. Finally, "Conclusions and future work" provides concluding comments.

Background knowledge

This section studies the methods available in this field. The attention is divided into two groups:

Recommender systems with LSTM

With the advancement of the artificial neural network, numerous research has explored using deep learning techniques for improving recommendation systems [20, 21]. CNN and RNN are two commonly used structures of deep learning. CNN is a form of artificial neural network that can detect information in different positions with excellent accuracy. This model has solved several problems in image processing and automatic

NLP, such as opinion analysis, text summary, and others. It is characterized by a particular architecture to facilitate learning. A CNN is a multilayer network so that the output of one layer will be the input of the next layer. It is usually composed of an input, one to several hidden layers, and an output [22].

Another artificial neural network is RNN which is an interconnected and interacting network of neurons, where the neurons are connected by arcs of a weight W. This type of network is very useful in the case of inputs of varying sizes, as well as for time series, namely automatic translation, automatic speech recognition, and automatic pattern recognition. The direction of propagation of the information in this type of artificial neural network is bidirectional; it keeps the sequence of data, and it can make the connection between an input of long sequences because it is based on a loop thanks to its internal memory [23].

LSTM is an extension of RNN able to solve the problem of the vanishing of the gradient thanks to its memory, which makes it possible to read, write and delete the data through three gates: the first allows or blocks the updates (Input Gate); the second disables a neuron if it is not important based on the weights learned by the algorithm, which determines its importance (Forget Gate); and the third is a control gate of the neuron state in the output (Output Gate) [12, 24].

In recent research, Li et al. [25] introduced a novel method for the sentiment classification of user reviews. The underlying idea is to make input data consistent in size and to improve the composition of sentiment information in the reviews. Their method uses a combined model comprising CNN and conventional LSTM. Experimental evaluation of the Stanford Sentiment Treebank dataset and tourism review dataset shows improvement over baselines of LSTM, among others [25].

Wang et al. [26] suggested a method for POI recommendation employing deep learning in LBSNs concerning privacy. First, user information, relationships, and location information are reviewed. Then, based on the history of the user and the order of check-in POIs, the LSTM mechanism is created, and the user information is used as input to obtain short-term and long-term user preferences. Finally, social network knowledge and semantic knowledge are placed in various input layers, and to recommend the next POI to users, temporal and spatial information of user histories are used [26].

In another paper, Sun et al. [27] introduced an LSTM method based on preference for the next POI suggestion. In general, the dynamic behavior of users is presented under the two headings of long-term and short-term preference. Long-term movements are usually repetitive and generally do not show adaptability; on the other hand, the short-term tends to be more variable. Their method, therefore, considers both long-term and short-term behaviors [27]. Zhao et al.

[28] proposed a spatiotemporal gated network for POI recommendation based on a gated mechanism that enhanced the long short-term memory (LSTM)-based RNN network by introducing gates to capture the spatiotemporal relationships between successive check-ins.

In another research, Bai et al. [29] proposed a deep learning model that employs tensor decomposition and regression to decompose the visual question-answering deep network. In their algorithm, CNN is compressed with LSTM to accelerate processing simultaneously. They proposed to conduct various decomposition methods and regression strategies on different Layers. This model is useful in addressing the problem of complex visual question-answering tasks where multiple modalities, such as text, images, and videos, are involved [29].

In one of the latest research endeavors, Zhuang et al. [30] conducted a research study addressing the nonuniform trial length problem that arises in practical applications of iterative learning control (ILC) for linear time-invariant multiple-input-multiple-output (MIMO) systems with input constraints. They propose an optimal ILC algorithm specifically designed for MIMO systems with nonuniform trial lengths, which incorporates a modification to the optimal ILC framework by utilizing the primal–dual interior point method to handle input constraints. The authors demonstrate the algorithm's effectiveness through mathematical expectation analysis, highlighting its monotonic convergence property. Numerical simulations using a mobile robot validate the algorithm's performance and its applicability to real-world scenarios [30].

Tao et al. [31] presented a PD-type ILC approach for systems with multiple time-delays and polytopic parameter uncertainty. Their study incorporated both time and trial domain objectives to robustly design the controller. The proposed approach offered advantages such as avoiding computation with large matrices required in alternative methods. By utilizing the generalized Kalman–Yakubovich–Popov lemma, the controller ensured monotonic trial-to-trial error convergence in the finite frequency domain, reducing conservatism associated with approaches covering the entire frequency range. The convergence conditions were expressed as linear matrix inequalities, enabling efficient algorithmic solutions. Numerical simulations confirmed the effectiveness of the proposed method and showcased its superior robust tracking performance when compared to other types of ILC [31].

Zhou et al. [32] introduced an extended framework of point-to-point ILC within discrete linear time-invariant (LTI) systems. The study utilized the tracking time instants of desired positions as changing variables, enabling the objective of minimizing energy while maintaining the required tracking accuracy. The multiobjective optimization problem was divided into two sub-problems, which were solved

using an iterative algorithm that combined the norm-optimal ILC approach with the coordinate descent method. The impact of model uncertainty on algorithm performance was also considered, and the algorithm was extended to handle constrained systems. It demonstrated robustness to model uncertainty and a certain level of robustness to output disturbances. The proposed algorithm was validated using a twin-rotor aerodynamic system (TRAS) model, confirming its effectiveness and applicability [32].

In contrast, our paper proposes a sequential neural recommendation system that utilizes BERT and LSTM on social media posts. The proposed model takes advantage of the contextualized representations provided by LSTM and the sequential dependencies captured by BERT to improve the quality of recommendations. The model is specifically designed for social media platforms, where users generate a large volume of data in the form of share pictures, posts, comments, and likes, and it can be used to recommend relevant content to users based on their preferences and behavior.

It is noteworthy that the ILC papers address the optimization of control algorithms for systems with input constraints, nonuniform trial lengths, and time delays, focusing on achieving robust performance and convergence properties. In contrast, our work utilizes BERT and LSTM models to capture sequential dependencies and contextualized representations of social media posts. Our objective is to enhance the quality of content recommendations based on users' preferences and behaviors. By establishing connections between the two areas, we can gain valuable insights for enhancing the performance and robustness of our sequential neural recommendation system within the social media domain.

Recommender systems with BERT

BERT is a contextualized representation model found by Google. The model is based on a deep bi-directional Transformer-encoder in which the Transformer uses parallel attention layers instead of sequential recurrence models. Both right and left contexts are appeared by using Transformer-encoder to pre-train better and stronger contextualized representations [14]. This recollection of previous input is critical for effective sequence learning. Transformer deep learning models, such as BERT, are an alternative to RNNs that utilize an attention technique to parse a sentence by focusing on the most relevant words before and after the sentence. Transformer-based deep learning models do not require sequential information processing, enabling significantly more parallelization and reduced training time [8].

An NLP application converts the input text into word vectors using techniques such as word embedding. With word embedding, each word in a sentence is converted into a series of numbers before being fed to RNN variants, Transformers, or BERT to understand the context. These numbers change

over time as the neural network trains itself, encoding unique properties such as the semantics and contextual data for each word so that similar words are close together and dissimilar words are far apart in this number space [33].

Most sequential recommendation systems rely on RNN and their variants, gated recurrent units, or long short-term memories [34]. These methods involve encoding records of historic data into vectors in a variety of ways to provide a representation. Transformer neural architectures and, particularly, the BERT language model have enabled Transformer-based sequential recommendation models to achieve optimal performance in predicting the next items, including NOVA-BERT [35] and DuoRec [36].

Among the early pioneer works, Islam and Bhattacharya [37] propose a combination of a BERT model using a collaborative recommendation architecture for good accuracy. Fan et al. [35] propose a sequential recommendation model that uses controlled bidirectional self-attention to model user behavior sequences in an augmented manner. They construct item interaction patterns based on the above-mentioned user behavior characteristics and then use these interaction patterns to augment attention locally. The item interaction patterns are created from a set of trainable parameter pairs, so it is learnable and lightweight. The sequential recommendation approach utilized in another study by Sun et al. [38] predicts the next item with which the user will interact. First, the model generates a list of the items the user has engaged with chronologically based on the historical sequence. Then, an embedding layer that includes position and input embedding is applied to each token within the item sequence. The study proposed by Yang et al. [39] uses BERT and Latent Dirichlet allocation to provide semi-supervised semantic-based recommendations of literature and researchers.

A model based on BERT and sequential recommendation proposed by Seol et al. [40] can enhance recommendation performance by using session tokens, embedding session segments, and time-aware self-attention in addition to taking advantage of session information while using minimal additional parameters.

Another method proposed by Zhuang and Kim [41] involved the BERT model based on review data and rating labels from the TripAdvisor site. The study proposes a multi-criteria system for recommending target customers suitable for the hotel. The proposed recommender uses fine-tuned BERT to predict the criteria ratings in the absence of adequate ratings for TripAdvisor's six criteria. For each hotel, a multicriteria recommender model suggests top-N customers according to these predicted ratings.

In their work, Wang et al. [42] introduced a novel POI recommender system that harnessed the deep semantic information within trajectories to enhance the trajectory embedding quality. They leveraged a pre-trained language model to extract implicit deep semantic information, enabling the

establishment of causal transfer constraints between check-ins through semantic means [42]. Thaipisutikul and Chen [43] presented a novel and enhanced deep sequential model, for POI recommendation. Their approach aimed to capture the user's short-term preference by utilizing a self-multi-head attentive aggregation layer. This layer effectively captures the intricate relationships between non-consecutive POIs, particularly in complex situations, and aggregates all POI representations to create a detailed and refined representation of the user's short-term preference [43].

Huang et al. [18] developed a method for aspect-based sentiment analysis and POI recommendation, aiming to accurately capture sentiment information from social media data with limited labeled data. Their approach utilizes the pre-training model BERT to obtain embedded word representations that combine semantic information from the text. Through contrastive learning, point clusters belonging to the same class in the embedded word space are brought closer together, while clusters from different classes are separated. The study also analyzes the relationship between comment ratings and their influence on user perception, determining the optimal performance formula for the loss function [18].

The proposed method in this paper offers several advantages over existing recommendation systems: first, the proposed method exploits the power of BERT and LSTM to capture both the semantic and sequential information in social media posts. This allows the model to provide more accurate and relevant recommendations to users by taking into account the context and sequence of their interactions with social media content. Second, the model is designed to provide personalized recommendations based on the user's preferences and behavior. This is achieved by leveraging the contextualized representations of LSTM and the sequential dependencies of BERT to identify patterns in the user's interactions with social media content. Furthermore, our method can be applied to a wide range of social media platforms and recommendation tasks, making it a versatile and scalable solution for recommendation systems. Finally, the proposed method is able to handle noisy and sparse data, which are common on social media platforms. This is achieved through the use of attention mechanisms that can identify relevant information in the input data and filter out irrelevant information.

The proposed method

Our method introduces a Neural hybrid trip RS for the tourist industry that utilizes tourist demographic, contextual, and geo-tagged information to suggest a list of places in a town. This method is separated into two phases, as illustrated in Fig. 1 (offline and online). Contextual data is derived from the image time and climate network after offline preprocessing

("Data preprocessing"). These data are added to the dataset to make it more complete ("Enriching geo-tagged photo with contextual information"). Moreover, clustering methods are being utilized to establish the region of interest, with each zone containing one or more POI relying on the geographic coordinates of the images.

Regarding that, places are found by repeating the clustering technique on the Area of Interested (AOI) results ("Finding POIs"). Now, in ("Finding POIs"), a profile is created for each of these POIs (L) by calculating their publicity and situational characteristics ("Producing the profile of points of interests (POIs)"). The User-POI ("User-POI detection") is computed with a weighted graph, and User–User ("LSTM calculation of user-user asymmetric schema") is calculated by a neural model that uses the hybrid similarity of opinions and interests based on user reviews and prior visits to each POI. Because user similarities are computed based on the similarity of ratings and reviews, a more comprehensive similarity is captured. These feature sets have databases where they maintain their data. At the next step, a BERT model is trained by finding prior user journeys in which sequences of POI were created based on the POI visits made by any user. BERT was trained with those ("Training process of BERT").

POI sequences were created using the visit time of any POI user ("Training process of BERT") to find prior user journeys. The POI sequence database was also used to hold these POI sequences. The findings were saved for use in the subsequent phase, accelerating processing and enhancing speed. Tourists can register for the framework during the online phase by supplying their demographic data, age, sex, town, nation, relationship, and profession. When a tourist queries the framework, the question is an argument with the context data, like geographic location (the current place) and climate data for the current travel dates ("Enriching user queries by contextual data"). Next, contextual pre-filtering is also utilized to choose POIs in that place based on the user's actual place ("Pre-filtering based on context").

Next, the similarity (L') between the current tourist and those who visited the filtered POIs is computed ("Combination of the recommendations"). Both the CF and DB approaches were used in our similarity. Tourists who are more similar to the current tourist are then selected. POIs can currently be picked and suggested to rely on the top-ranking neighbor's tourists. Before producing recommendations, the user's geographical proximity and the current climate are assessed (contextual modeling). The expected list of POIs was then selected ("Recommendation").

BERT is applied in this paper to discover sequential POIs. Top-N POIs are fed into this algorithm, and the next POI is predicted based on them. BERT then predicts the next POI based on this sequence (with two POI). Each sequence
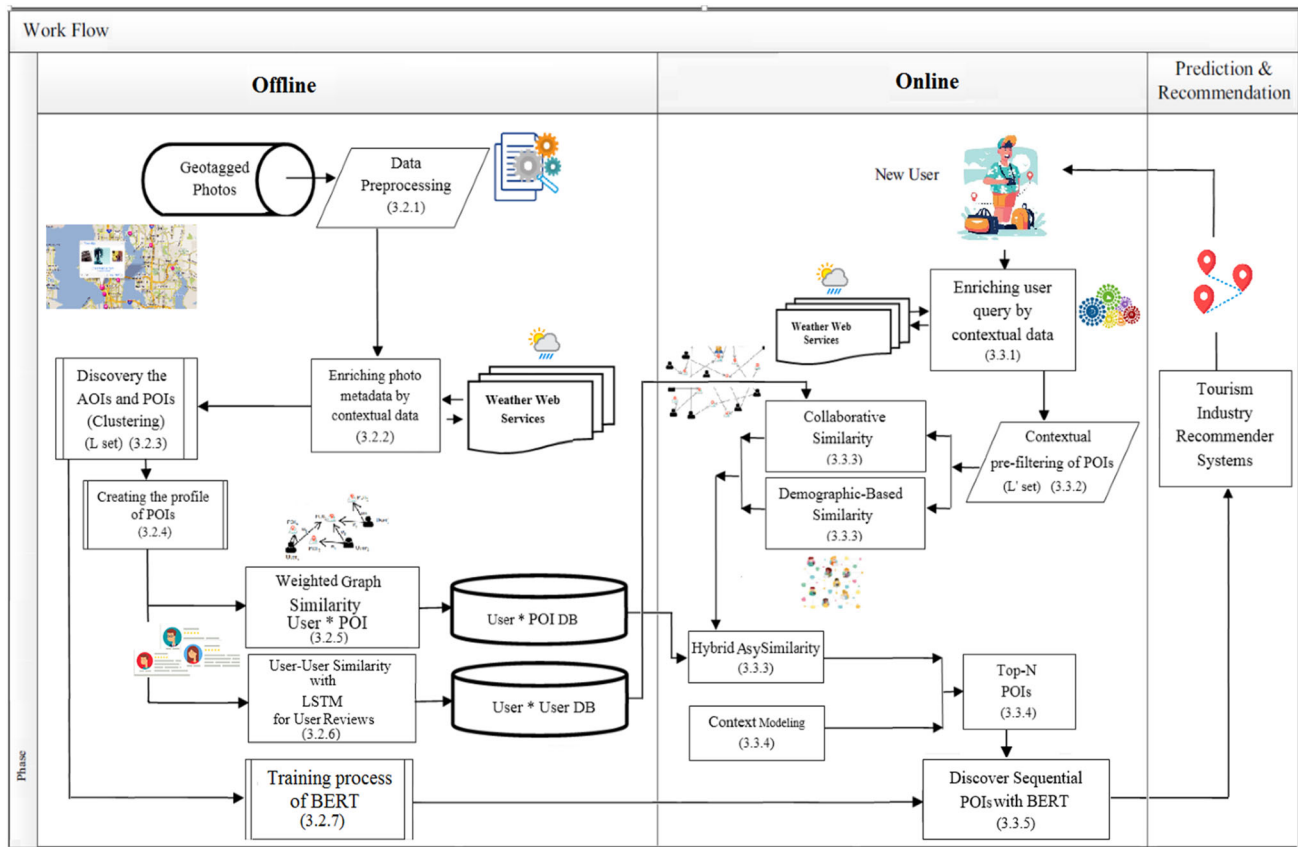
Fig. 1 The workflow of our method

can have up to POIs ("Discover sequential POIs with BERT stage"), where is the length of the sequence.

The Top-N trip sequential patterns are applicant journey patterns. Finally, the target tourist is provided with travel suggestions based on current contexts, demographic factors, and sequential patterns of movement.

The steps that follow go over the contribution of the proposed framework as well as the algorithms that were used to create them.

Problem identification

The following is a definition of the challenge of proposing exciting tourist destinations and POI sequences in geo-tagged social networking sites: $P = P_1,\ldots, P_n$ is a series of publicly accessible geo-tagged photographs that demonstrate the approach of locating tourist destinations in a city, assessing their attractiveness, and providing intriguing journey sequence suggestions based on prior tourist journeys and travel sequence patterns. Interestingly, travelers' publicity image collections are being utilized to offer enticing tourism locations and intriguing tourism sequences based on the visitors' present context.

In other words, the research question of this study is: how can a context-aware recommender system be developed for personalized tourist destination recommendations using contextual and demographic data, user reviews, and geo-tagged social network photos? The research objective is to propose a novel hybrid RS that integrates contextual and demographic data, user reviews, and geo-tagged social network photos to provide personalized tourist destination recommendations and POI sequences based on prior tourist journeys and travel sequence patterns.

Offline phase

Some calculations were conducted offline, and the data obtained was preserved to improve the speed of our framework.

Data preprocessing

First, the data source has to be cleaned and preprocessed due to the inclusion of some unclear and unsuitable data. This included deleting unclear information and images with insufficient parameters. It is important to mention that while visiting a POI, a person can photograph it many times. As

Table 1 Context matching

Time context	Visit day	Saturday, Sunday	Weekend
		Monday, …, Friday	Working day
	Visit time	06:00–12:00	Morning
		12:00–18:00	Afternoon
		18:00–06:00	Night
	Visit season	March, April, May	Spring
		June, July, August	Summer
		September, October, November	Fall
		December, January, February	Winter
Weather context	Temperature	> 34 °C	Hot
		18–34 °C	Warm
		< 18 °C	Cold
	Weather	Sunny, clear sky	Sunny
		Cloudy, broken clouds, scattered clouds	Cloudy
		Rain, fog	Rainy
		Snow, snowfall	Snowy

long as the time change between a person's initial and subsequent images is smaller than a threshold, both photos are considered one and pertain to the same visited place.

Enriching geo-tagged photo with contextual information

This stage generated and saved contextual data for the image elements in the dataset. This data, including the time and geographic place, is paired with each image posted by users. In line with the map in Table 1, the current climate is derived using the climate application. Contextual information such as climate, temperature, season, and other date information is included in the database.

Finding POIs

The DBSCAN approach was used to cluster geo-tagged pictures, and distance measures obtained from the Manhattan distance were used to extract spatial positions. This approach offers significant benefits over previous clustering algorithms, including the ability to discriminate clusters using arbitrary areas [44] and the demand for little scope information to determine the parameters. It also does a good job of grouping vast amounts of data. The thickness point for clusters in the DBSCAN is the same. The minimum number of points required to form a cluster (MinPts) and the

radius (Eps) are crucial parameters. The size and density of clustered places can vary. AOIs were extracted from a batch of geo-tagged pictures using the DBSCAN clustering technique. After that, the algorithm was run again with the proper settings on the observed AOIs. Consequently, a collection of POIs was created, as well as a database of important tourist places (L).

Producing the profile of points of interests (POIs)

To evaluate the publicity and context features of each POI, Eqs. (1) and (2) were used to create a profile of the discovered POIs.

$$\text{PlacePopulartity (POI)} = \log\left(\frac{N}{N_l}\right). \tag{1}$$

Here N_l represents a specific POI set of visits from a region, whereas N represents the overall number of visitors from that region.

A new weighting context vector structure is presented in our work as $\vec{C_l} = \langle c_{(l,1)}, \ldots, c_{(l,k)}\rangle$, here $c_{(l,j)}$ indicates the context (j) of any POI, and (n) represents the sum of contextual factors (Table 1). $c_{(\text{POI},j)}$ is calculated utilizing the TF-IDF algorithm formed on Eq. (2).

$$c_{(\text{POI},\,j)} = \text{TF}_{\text{POI}} * \text{IDF}_{\text{POI}} = \frac{w_{(\text{POI},\,j)}}{w_{(0,\,j)}} * \log\frac{w_{(0,0)}}{w_{(\text{POI},0)}}. \tag{2}$$

Here $w_{(\text{POI},j)}$ shows the number of visitors from the POI in context (j), $w_{(0,j)}$ indicates the sum of travels in context (j) from all POIs in the present town, $w_{(0,0)}$ represents the number of the journey in any context from all POIs in the present town, and $w_{(\text{POI},0)}$ indicates the number of the journey in each context from the POIs.

User-POI detection

The similarity between User and POI is used to construct a weighted undirected graph $\text{Graph}_{\text{User-POI}} = (\text{User; POI; Edge}_{\text{User-POI}}; \text{Weight}_{\text{User-POI}})$ to identify the preferences of a group of tourists U inside a collection of places L. $\text{Edge}_{\text{User-POI}}$, and $\text{Weight}_{\text{User-POI}}$ represents collections of sides and side weights among the User and POI, respectively, reflecting users' visitations and the instances of visitations to a specific POI.

Figure 2 illustrates the similarity between users and POIs. Suppose n users and m POIs, an n by m adjacency matrix $\text{Matrix}_{\text{User-POI}}$ ($\text{Matrix}_{\text{User-POI}} = [T_{ij}]$) is created for the network $\text{Graph}_{\text{User-POI}}$, where T_{ij} denotes the instances the jth POI has been visited by the ith user. If $T_{ij} = 0$, it indicates that the ith tourist has never seen the jth POI.
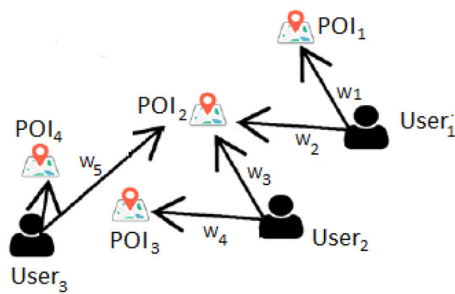
Fig. 2 The relation between Users and POIs

LSTM calculation of user–user asymmetric schema

The resemblance among tourists who had been before the places were assessed and saved for utilization throughout the online phase. The experiences of other tourists were utilized to provide recommendations for the intended client. Equation 4, a modified version of Sorensen's formula (Eq. 3), was utilized to achieve this result. The goal would have been to estimate tourist similarities relying on the number of photographs they saw and to develop a rational link between their attractiveness.

The suggested method is based on the concept that when two users take images of places that are less popular and visited by fewer people, this similarity is much higher than that of the other. Two visitors who visit a fewer part of a town are more interested in sharing similar concerns. Consequently, while computing similarity, the suggested similarity metric tries to account for individuals' preferences for commonly visited sites.

$$\text{Sim}(u, \text{v}) = \frac{2 * |l_u \cap l_v|}{|l_u| + |l_v|} \tag{3}$$

$$\text{Sim}_{PP}(u, \text{v}) = \frac{2 * \sum_{l \in l_u \cap l_v} \text{PlacePopulartity}(l)}{\sum_{l \in l_u} \text{PlacePopulartity}(l) + \sum_{l \in l_v} \text{PlacePopulartity}(l)}. \tag{4}$$

Here l_u represents the places visited by user u, and PlacePopulartity(l) indicates the popularity of a place (l).

Now, the similarity between two users based on collaborative filtering can be calculated using Eq. (5):

$$\text{Sim}_{CF}(u, v) = \text{Max} \left(\text{Sim}_{PP}(u, v) \quad \text{and} \quad \text{Sim}_{Review}(u, v) \right). \tag{5}$$

In this research paper, our objective is to enhance the estimation of user-user similarity within a CF framework by leveraging users' reviews. The proposed model aims to incorporate user reviews as additional information to improve the accuracy of suggesting different POIs to users. Reviews written by users have been identified as highly effective indicators for predicting user interest. By integrating this feature into Eq. (5), we can enhance the performance of the recommender

system. In other words, incorporating users' feedback information enables us to capture their interests based on their opinions and the aspects of POIs that hold significance for them. Therefore, our study focuses on utilizing users' reviews to provide a more precise estimation of user-user similarity, ultimately leading to enhanced recommendations for various POI.

The $\text{Sim}_{Review}(u, v)$ metric measures the similarity of user reviews based on their textual similarity. To be more precise, in the second section of Eq. (5), we need to utilize a text similarity method to calculate the similarity between user reviews. By calculating the similarity of texts, we can determine the similarity between two users based on the maximum similarity among all pairs of their reviews, as demonstrated in Eq. (5). Notably, no preprocessing has been conducted on the reviews in this introduced method; consequently, the outcomes are provided without any language-specific preprocessing and can be used for each language [45, 46].

In this study, we implemented a neural document representation method aimed at constructing vector representations for entire documents. To consider the dependency of input data and facilitate information persistence, we utilized RNNs, specifically LSTM networks. RNNs are capable of capturing dependencies, but they struggle to retain long-term dependencies. To overcome this limitation, LSTM networks were introduced [12, 23].

To exploit the benefits of this network, we trained an encoder-decoder neural network with two LSTMs—one in the encoder and one in the decoder. This architecture allowed us to effectively capture and generate meaningful representations of input sequences.

In our model, the encoder LSTM takes the input sequence, such as a review, and processes it step by step. At each time step, the encoder LSTM analyzes the input and updates its hidden state based on the current input and the previous hidden state. This recurrent nature of the LSTM enables it to retain and propagate information from earlier time steps, allowing it to capture the sequential dependencies within the input sequence.

Once the input sequence has been processed by the encoder LSTM, the final hidden state of the encoder LSTM serves as a condensed representation of the entire input sequence. This fixed-size vector, also known as the context vector or latent representation, encapsulates the essential information from the input sequence.

The decoder LSTM receives the context vector as its initial hidden state and is responsible for generating the output sequence. It operates in a similar fashion to the encoder LSTM but in reverse. At each time step, the decoder LSTM takes the previous output (either the predicted token or the ground truth token during training) and its hidden state from the previous time step, and generates the next output token.

This process is repeated until the desired output sequence is generated.

By training this encoder-decoder model with two LSTMs, we effectively capture the relevant features and dependencies within the input sequence and generate coherent and meaningful output sequences. The use of a fixed-size vector representation allows for efficient handling of varying-length input sequences, making it easier to model and manipulate reviews in a consistent and manageable manner.

In (Eq. (6)), where u and v are two users, u_i is the ith text review of user u, and v_i is the ith text review of the user v.

$$\text{Sim}_{\text{review}}(u,\ v) = \frac{\sum_{i=1}^{n} \vec{V}_{u_i} W_{v_i}}{\sqrt{\sum_{i=1}^{n} \vec{V}_{u_i}^{2}} \sqrt{\sum_{i=1}^{n} \vec{W}_{v_i}^{2}}}. \tag{6}$$

In the actual world, user similarities are not always symmetrical and may not be identical. Most conventional measures of similarity value the similarity relationship between two users equally. These strategies are based on the concept that $\text{sim}(u,\ v)$ equals $\text{sim}(v,u)$; however, the impact of two different users on one another differs, hence an asymmetric schema is used to traditional similarities in CF approaches to get a very realistic similarity [47]. This work uses asymmetric schema to bypass the limitation. The rate of similarity places among tourists, adjusted by the figures of places assessed by the present tourist, Eq. (7), is used to create the asymmetric similarity measure.

$$\text{sim}_{\text{Asy - Measure}}(u,\ v) = (1 - \exp(-|l_u \cap l_v|)/|l_u|). \tag{7}$$

Here l_u represents the number of POIs visited by u. This equivalence looks at the proportion of common ratings that users have among all of their rated things rather than the proportion of common ratings in the total number of ratings among tourists. As a result, Eq. (8) contains this parameter.

$$\text{Sim}_{\text{AsyCF}}(u,\ v) = \text{Sim}_{\text{Asy - Measure}}(u,\ v) * \text{Sim}_{\text{CF}}(u,\ v). \tag{8}$$

We should also consider the preferences of each visitor. Different tourists have different tastes. We utilize the median of the PlacePopularty to represent the user preference to display this conduct distinction. The following is a representation of the user PlacePopulartyPreference (UPP) based on similarity metrics:

$$\text{Sim}_{\text{UPP}}(u,\ v) = \frac{e^{-(|r_{u,p}-r_{\text{med}}|*|r_{v,p}-r_{\text{med}}|)}}{\left[1 + e^{-(|r_{u,p}-r_{\text{med}}|*|r_{v,p}-r_{\text{med}}|)}\right]^{2}}, \tag{9}$$

where $r_{u,p}$ indicates the rating of PlacePopulartity by user u. The r_{med} represents the median value of two tourists, u and v, on a rating scale.

By combining Eqs. (14) and (15), it is possible to arrive at the new formalization, which we refer to as the improved new CF asymmetric similarity model (AsyNCF) (Eq. (10)). Hybrid RSs enhance performance by combining multiple recommendation techniques. CF is usually employed in conjunction with another method to avoid the ramp-up issue. We utilized the feature combination method with multiplication since this hybrid includes two distinct recommendation components: the contributor (in our study, UPP) and the genuine recommender (in this study, Asymmetric CF). Thus, the relationships between the product's components have been maintained. The authentic recommender operates on data modified by the contributor.

$$\text{Sim}_{\text{AsyNCF}}(u,\ v) = \text{Sim}_{\text{UPP}}(u,\ v) * \text{Sim}_{\text{AsyCF}}(u,\ v). \tag{10}$$

Training process of BERT

Initially, This stage extracts the place sequences to determine the tourist travels, relying on the place visit order. The period of every user's trips to POIs is also considered. A single trip is formed when the time variation among two sequential POI visits is lower than a threshold level; these distinct journeys are formed as long as the time variation is higher than the threshold level. We utilize an 8-h threshold in our strategy, as in earlier studies [48]. A factor is used to track the periodicity of each trip. The number of users that visited each trip is used to establish the sequencing frequency in this approach. Each journey has its collection of POIs and its own set of POI orders.

The sequential recommendation process is inspired by the deep bidirectional self-attention model BERT applied for language modeling. Figure 3 shows the BERT model with three layers, which are explained in the following [49]:

(A) Embedding layer: With this layer, inputs are learned from the journey, which consists of POIs, each with information including POI ID, topic, PlacePopularity (POI), and contextual information. The embedding layer's output is fed into the Transformer layer, which is the next layer.
(B) Transformer layer: Transformer blocks are arranged in $\mathscr{L}$ layers. In each transformer block, a list of embeddings for tokens is taken in, and an identical number of embeddings is produced on the output (while features change because of the transformation). In the last Transformer block, the output goes to the projection layer, which is the next layer.
(C) Projection layer: In this layer, for prediction, a softmax layer is applied to project the learned hidden representation from the previous layer onto the item space. A
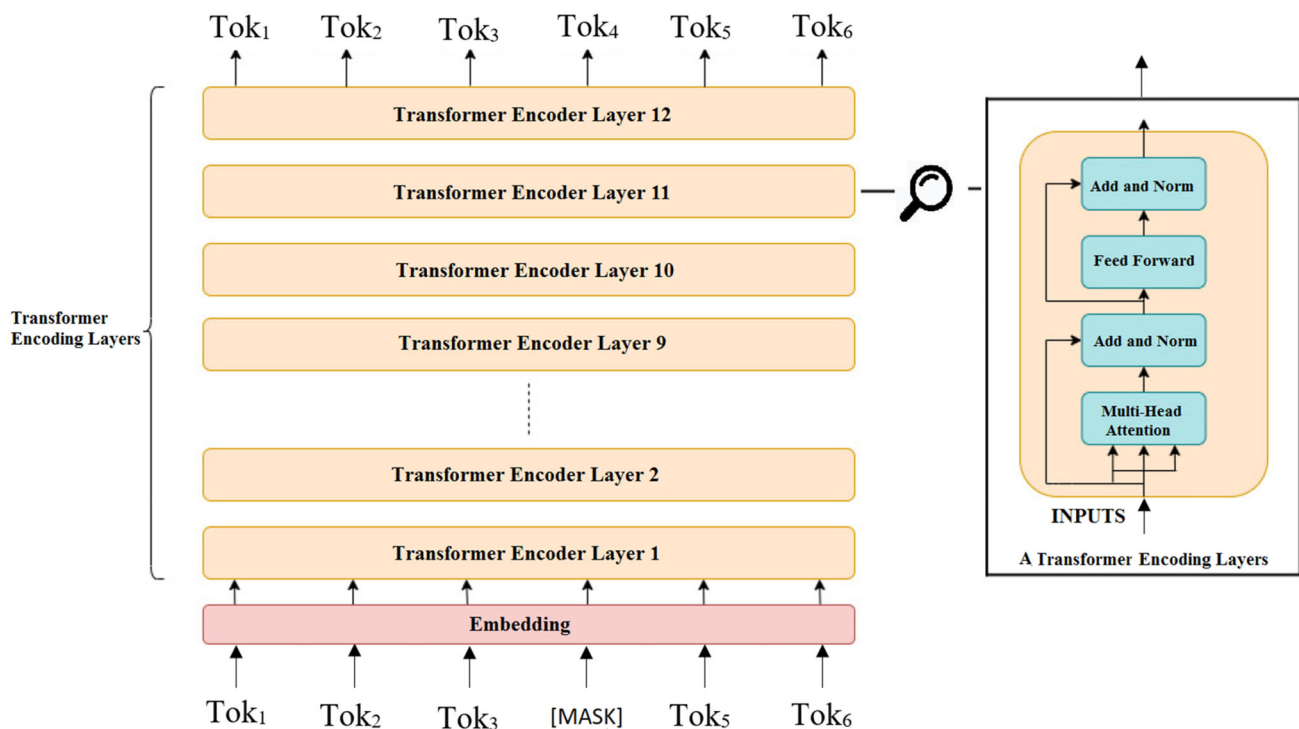
Fig. 3 BERT model architecture

Cloze task is used for training, in which the model predicts masked items in the sequence of interactions [50].

BERT can be trained by Bayesian Personalized Ranking (BPR) [51]. The BPR is a pairwise personalized ranking loss based on the maximum posterior estimate. The training task is optimized using BPR pairwise, maximizing the difference between negative and positive pairs. Generally, BPR assumes that users prefer positive items over non-observed or negative items.

There is an obvious correlation between the training model and the correct values predicted by the proposed dataset. As a result, the proposed model learns both the useful representations for each POI and the key patterns between them.

It is noteworthy that the maximum sequence length a BERT model usually takes is 512 tokens [52, 53]. This proposed method for handling this potential limiting factor of BERT by truncating he input sequences to a maximum length of 512 tokens.

Furthermore, to ensure that we do not lose important information from the truncated sequences, we also implement a sliding window technique, where we divide the long input sequences into shorter segments and pass each segment through the BERT model separately. This allows us to capture more information from the longer sequences, while still adhering to the 512-token limit of the BERT model.

In this research, we used the pre-trained BERT base uncased model released by Google, which was trained on a large corpus of general text data from the English language. In addition, we fine-tuned the BERT for another scenario on our specific task of next-POI recommendation. For fine-tuning, we first initialized the BERT model with the pre-trained weights and then fine-tuned it on our dataset using a sequence-to-sequence learning approach. This involves feeding the input sequences of historical sequential POIs of trip users to the BERT model and training it to predict the next POI in the sequence.

Online phase

The following steps are included in this stage. Our method answers the target user's request quickly and interactively.

Enriching user queries by contextual data

During the online phase, the system computed the time requested by the user to visit as the tourist's desired time. The weather and temperature contexts for that location were then provided and completed following the mapping in Table 1 of the user context inquiry, leveraging the weather web service's season and visit time contexts. For the current user, a structure of context factors such as $(\vec{V}_u)$ is constructed. If a context condition is met, it is assigned a value of one; otherwise, it is assigned a non-value.

Pre-filtering based on context

The data for that city was chosen in this stage based on the current user's geographical attributes in the enhanced query.

$$\text{Pred}(u, l) = \frac{\sum_{v \in U'} \text{Sim}_{\text{WT-context}}(C_u, C_l) * \text{Sim}_{\text{loc-context}}(l_u, l_l) * \text{Sim}_{\text{AsyHybrid}}(u, v) * (r_{v_l})}{\sum_{v \in U'} \text{Sim}_{\text{AsyHybrid}}(u, v)}, \tag{13}$$

This contextual pre-filtering creates the collection of those city locations (L').

Combination of the recommendations

This phase uses Eq. (11) to compute the hybrid similarity. In terms of the present tourist and those who visited the set (L') of places, this equation highlights the similarities between CF and DB.

$$\text{Sim}_{\text{AsyHybrid}}(u, v) = (1 - \beta) * \text{Sim}_{\text{DB}}(u, v) + (\beta)\text{Sim}_{\text{AsyNCF}}(u, v). \tag{11}$$

This compound is balanced using the coefficient (β) to smooth out the linear connection.

Equation (12) was used to estimate the demographic similarities between the two tourists.

$$\text{Sim}_{\text{DB}}(u, v) = \frac{|\text{num}_1\left(\vec{D}_u \cap \vec{D}_v\right)|}{|\text{Demograpic FeatureVector}|} * 1/\left(1 + \frac{|\text{age}_u - \text{age}_v|}{\max(\text{age}) - \min(\text{age})}\right), \tag{12}$$

For each user, a demographic characteristic (excluding age) vector such as ($\vec{D}_u$.) is created. The first tourist demographic characteristic vector is compared to the second tourist demographic information vector when comparing users based on their demographic features. If two users have the same value for a certain property, such as sex, the value of one is utilized. Using the $\text{num}_1(\vec{D}_u \cap \vec{D}_v)$ function, the number of units in the two users' common factors vector is tallied and divided by the number of demographic factors examined by the users. The output result of this similarity is always between zero and one.

ven the importance we place on the aged character, we utilized the tourist age as a distance attribute model where age_u and age_v are the ages of the two tourists, u and v.

Following that, utilizing the similarity metric presented in this equation, the present tourist's similarity to other tourists visiting the aid area (user–user) is determined. These findings are used to choose people who have visited that city with a greater similarity score to the present user.

Recommendation

The level of the intention of the present tourist u to visit destinations can be determined using Eq. (13) based on the similarity among tourists.

where (r_{v_l}) indicates the real rating of a tourist (v) for the place (l). In this equation, when computing the place (l) score relies on the tourist visit, $\text{Sim}_{\text{WT - context}}(C_u, C_l)$ and $\text{Sim}_{\text{loc - context}}(l_u, l_l)$ is used as a weight.

The context factor $\text{Sim}_{\text{loc - context}}(l_u, l_l)$ is the next to be evaluated. The farther a person is from a tourist site, the less likely they are to attend, and therefore the less suggested the attraction will be. The Manhattan distance was used to get the distance factor for the site (Eq. (14)).

$$\text{Distance}_{\text{Geo}(l_u, l_i)} = (|x_1 - x_2|) + (|y_1 - y_2|), \tag{14}$$

where we have the target user's geographical location $l_u(x_1, y_1)$ and the tourist location $l_i(x_2, y_2)$. To cover all points and achieve the closeness of distance, we utilize the double Laplace distribution equation (Eq. (15)).

$$\text{Sim}_{\text{Loc - Context}}(l_u, l_l) = \frac{1}{2\mu} * e^{\left(-\left|\text{Distance}_{\text{Geo}(l_u, l_l)}\right|/\mu\right)}. \tag{15}$$

The μ coefficient is utilized to convert the decrease rate in this case. The longer the space between the tourist's present place and the previously visited place, the fewer suggestions are offered.

Another context aspect examined by this method is the similarity of the climate and time ($\text{Sim}_{\text{WT - context}}(C_u, C_{l_i})$). During the offline process, a profile of POIs was built, and the vector form of dual factor values for every POI was stored.

Apart from that, contextual data was applied to the existing user query in the pattern of a vector ($\vec{V}_u$). in compliance with "Enriching user queries by contextual data", and on the other hand, having the vector template of the contextual metrics weight of the POIs ($\vec{W}_l$) enables the determination of similarity via the adjusted cosine formula (Eq. (16)).

$$\text{Sim}_{\text{WT - context}}\left(\vec{V}_u, \vec{W}_l\right) = \frac{\vec{V}_u * \vec{W}_l}{\left|\left|\vec{V}_u\right|\right| * \left|\left|\vec{W}_l\right|\right|}. \tag{16}$$

For the existing user context and every Location context, a context vector modeling such as ($\vec{C}_u$.) and ($\vec{C}_l$) is created. The output data of this similarity is always between zero

and one. The list items are sorted according to the projected points for each site, which are related tourist spots.

Discover sequential POIs with BERT stage

In this paper, BERT is used to identify sequential POIs. Initially, the algorithm will be fed the Top-N POIs to predict the next POI. Using these two POI (the Top-N and the predicted one), a third POI will be predicted based on this sequence in the subsequent step (the previous two POIs). This cycle must be repeated times, where is the sequence's length. In addition, this process will be repeated for the second Top-N POI to generate the second suggested trip following the production of this -length sequence. Note that as part of the Masked Language Modeling (MLM) task, certain POIs must be replaced with a token (MASK) to predict the next POI [54].

Simulations and experimental evaluation

In this part, many tests were conducted to show the effectiveness of the suggested strategy. We will first review the experimental datasets and model parameters in this regard. The evaluation measures are discussed after that. Following that, the experiments and their results are reported and debated. Ultimately, the data are evaluated and compared to findings acquired utilizing other cutting-edge approaches, such as Yelp and Tripadvisor Data sources.

MinPts, Eps for DBSCAN, and parameter β in Eq. (11) are method hyper-parameters whose values were investigated to determine how they affect the method's performance. In "Evaluation dataset", the clustering variables (MinPts, Eps) are identified and presented in Figs. 4 and 5. The weight of each DB and CF similarity between two users is then determined by the variable (β) in Eq. (11) in "Impact of parameter β", and the result is shown in Fig. 6.

Furthermore, the number of layers in a BERT network can be considered one of the hyperparameters for fine-tuning. To achieve effective fine-tuning of BERT, it is crucial to carefully select the number of layers in the network and make appropriate adjustments during the fine-tuning process. It should be noted that the remaining BERT parameters are kept unchanged and set to their default values. The results of this experimentation are discussed and presented in Sect. "Impact of layer count on fine-tuning BERT performance", and a visual representation can be found in Fig. 16.

The selection of the optimal coefficients was based on comparing the performance of the different configurations on a validation dataset. We assessed metrics such as precision, recall, F-Score, root mean square error (RMSE), and normalized discounted cumulative gain (nDCG). The configuration that yielded the best performance according to the chosen metric was selected as the optimal choice for the BERT model.

Evaluation dataset

This study employed Yelp, which can be found on the Yelp website "https://www.yelp.com/dataset." As a social network platform, this website has gained popularity. The data set contains 1.3 M reviews and 468 K tips by 340K users gathered from 2018 to 2021, where most of the reviews are made in North America. The Application Programming Interface methods were utilized for information. Table 2 illustrates different fields of the Yelp dataset.

As described in "Finding POIs", the offline state utilized the DBSCAN two-level clustering algorithm to discover user destinations from the dataset. We studied the DBSCAN settings and demonstrated how the number of recognized clusters varies when MinPts and Eps are altered. Correctly setting the technique's radius and minimum sample point parameters significantly impacts the method's accuracy. The size and concentration of clustered locations might vary.

The DBSCAN settings can be modified; therefore, it is important to look them over carefully to figure out how many regions there are. In this situation, the test approach was used to discover them. Figures 4 and 5 show how the number of recognized clusters changes as these two parameters' values alter.

The minimum sample size is a falling graph pattern when the radius reaches 120 for all attribute values, as seen in Fig. 4. For the parameter specifying the minimum sample sizes, the chart's declining slope is decreased to a parameter of 10 in Fig. 5. The clustering variables (Eps = 120, MinPts = 10) are put to these two parameters depending on the results, resulting in a total of 36 clusters. This data was separated into two non-overlapping halves to conduct the assessment, with 75 percent utilized for framework training and 25% for analyzing process.

To fine-tune BERT in our specific scenario, we leveraged the Python-based PyTorch library. This open-source deep learning framework has gained considerable recognition for its capability in both pretraining and fine-tuning BERT models.

We experimented with various parameter values for every formula. Therefore, the best results were obtained with this description of the report parameter values (Eq. (11): $\beta = 0.6$; Eq. (15): $\mu = 0.5$).

The second dataset used was the Tripadvisor dataset. This dataset is available at "http://www.Tripadvisor.com." We choose Tripadvisor as a real-world tourism dataset. In detail, this dataset contains 30,026 Users, 11,231 POIs, and 357,012 Reviews.

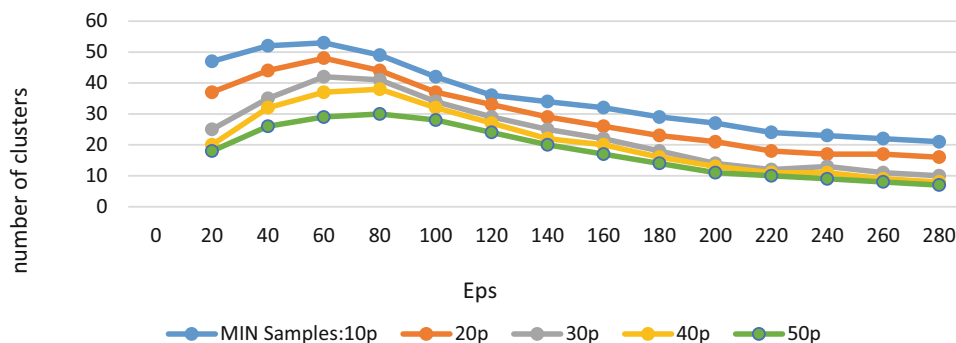**Fig. 4** Depending on the number of clusters discovered according to the Eps

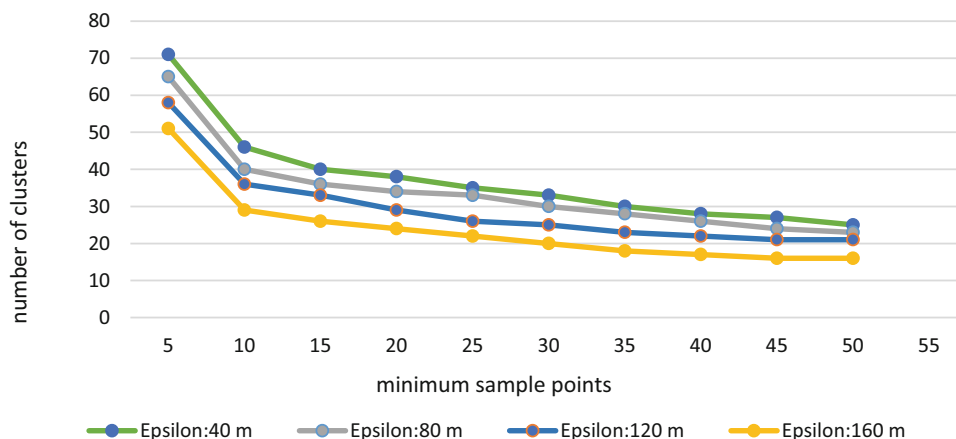**Fig. 5** Depending on the number of clusters discovered according to MinPts

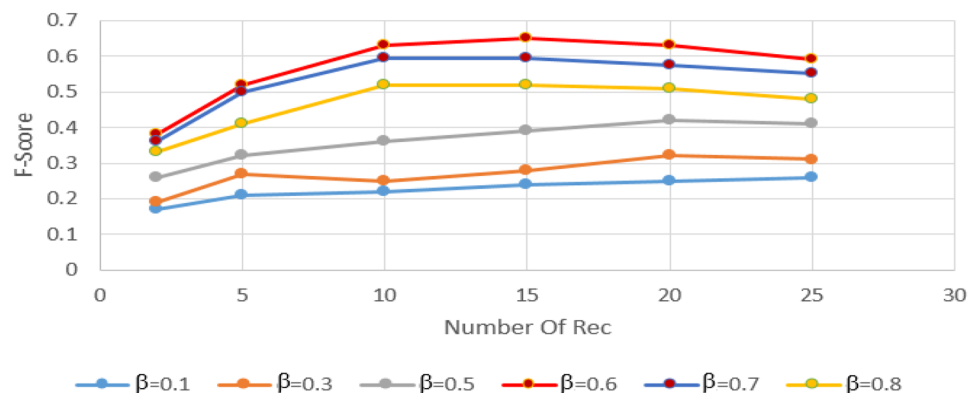**Fig. 6** The impact of DB and CF according to F-Score

Table 2 Fields of the Yelp dataset

Business ID	Geo-location	Stars	Text	Date	Tag
ID	POI geo-location	1–5 stars rating	Review text	Date	User tag

The evaluation metrics

The proposed method has been evaluated based on some criteria. In our study, we have selected precision, recall, F-Score, nDCG, RMSE, and MAP as evaluation metrics because they are widely used in the field of recommendation systems and information retrieval. Precision and recall measure the ability of the model to recommend relevant items

to users while minimizing irrelevant recommendations. F-Score provides a harmonic mean of precision and recall, which gives an overall view of the model's performance. nDCG is a ranking evaluation metric that measures the relevance of recommended items by taking into account their position in the ranked list. RMSE is commonly used to evaluate the accuracy of a model's rating predictions. MAP is a ranking evaluation metric that measures the average precision of a model's recommendations at different positions in

the ranked list. We have chosen these metrics based on their relevance to the task of sequential recommendation and their ability to provide a comprehensive evaluation of the model's performance.

Recall measure is presented as the ratio of accurate things offered to the whole number of relevant objects for the target user (Eq. 17).

$$\text{Recall} = \frac{\#(\text{Number of Accurate predictions})}{\#(\text{Number of Relevant Objects})}. \tag{17}$$

As demonstrated in Eq. (18), Precision is presented below.

$$\text{Precision} = \frac{\#(\text{Number of Accurate Predictions})}{\#(\text{Number of Total Predictions})}. \tag{18}$$

Equation (19) determines the Average Precision metric AP@N, an equation that calculates accuracy for all users.

$$\text{AP@N} = \frac{\sum_{k=1}^{N}(\text{Precision@}k * \text{Relevant}_k)}{M}, \tag{19}$$

(M) is the relevant item, and (Relevant_k) is an index role $(\text{Relevant}_k = 1$ if the item (k) on the recommended list is a related POI, that anyway $\text{Relevant}_k = 0)$.

Equation (20) describes The Mean Average Precision metric for (m) users.

$$\text{MAP@}m = \frac{\sum_{k=1}^{m} \text{AP}(u)}{m}. \tag{20}$$

RMSE highlights bigger absolute error levels (Eq. (21)) [55].

$$\text{RMSE} = \sqrt{\frac{\sum_{(u,i)|R_{u,i}}(\hat{r}_{u,i} - r_{u,i})^2}{N}}. \tag{21}$$

The user (u) and place (i) methods forecast the score value as $(\hat{r}_{u,i})$. The genuine value of the user's (u) rating for a place (i) is $(r_{u,i})$, and the total number of examined places is (N).

The F-Score is identified as Eq. (22) [56].

$$F\text{ - Score} = \frac{2 * \text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}}. \tag{22}$$

The projected suggestions' ranking efficiency is compared utilizing the nDCG (Eq. (23)). The more related topics of attention shown at the head of the proposed list, the higher the nDCG score [57].

$$\text{NDCG} = \frac{\text{DCG}}{i\text{DCG}} = \frac{\sum_{i=1}^{p} \frac{2^{\text{rel}_i - 1}}{\log_2^{(i+1)}}}{\sum_{i=1}^{|\text{rel}_p|} \frac{2^{\text{rel}_i - 1}}{\log_2^{(i+1)}}}. \tag{23}$$

Table 3 Comparison methods

Methods	#References	Description
(CF)	[58]	CF
(PR)	[59]	Public popularity
(Pre_CA-CF)	[60]	Contextual pre-filtering similarity measure
(CA-CF)	[61]	Jaccard measure
(ACA-CF)	[58]	CF + CA + asymmetric schema
(LSTM-CA)	[62]	CA + LSTM + sequence
(HBERT4Rec)	[63]	Content-based + CF + BERT
(ADBCACF)	[64]	Asymmetric CF and demographic data
(SeqHybrid)	[19]	Our previous work describes a sequential recommender system that combines context-awareness, demographic-based, and asymmetric CF

Here rel_i stands for the element ranked at a place (i), also rel_p stands for the list of related items in the relevant group in place (p).

Comparison approaches

Throughout this part, we evaluate our model with the following methods in Table 3.

Experimental results

The influence of CF and DB on suggestion accuracy is discussed first in this paragraph. The impact of neighborhood numbers on suggestion quality is then investigated.

The next sections compare the efficiency of our BERT-LSTM-hybrid technique to previous techniques that solve the cold start and data sparsity challenges based on MAP, Precision, Recall, RMSE, F-Score, and nDCG criteria.

Impact of parameter β

The weight of each DB and CF similarity between two tourists is determined by the variable (β) in Eq. (11). Consequently, the value selected for this factor has a substantial impact on the performance of the current plan. Figure 6 illustrates the results of the trials about the F-Score that was used to establish the optimal value for the parameter (β). The chart with a figure of $\beta = 0.6$ fared the best among the other figures of variable β, as can be observed. Consequently, setting this factor's starting figure to 0.6 is a good decision. As a result, the impact of neighbors on tourism suggestions takes precedence over demographic data.
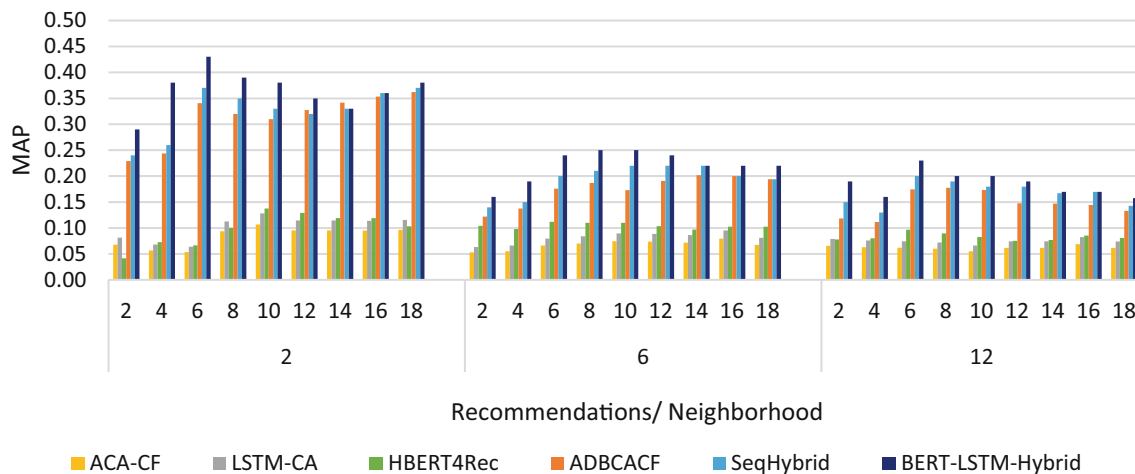
Fig. 7 The list of recommendations is used to compare BERT-LSTM-hybrid to others concerning the MAP metric (Yelp)

The effect of the neighborhood numbers

By proposing 2, 6, and 12 POIs among all president's suggestions, we studied and applied the influence of neighbor size on the accuracy of anticipated suggestions for neighbors of different sized. In this scenario, the number of neighbors grew between two and eighteen. Concerning MAP, the findings are shown in Fig. 7; when the number of suggestions surpassed six, adding POIs dramatically lowered the validity of the recommendation results when using MAP, according to these studies.

Users' preferences fluctuate, and they like to visit no more than six places in each region during their journey. As a result, recommending two to six POIs for tailored suggestions appears reasonable. Because of clustering, context data, user review, and demographic information, BERT-LSTM-hybrid exhibited a greater MAP score than previous approaches. According to the findings, asymmetric strategies outperform symmetric techniques and also tourist reviews of tourist trip records. Greater neighbors for the present tourist may be located because of the suggested strategy, and the POIs produced by such neighbors are more precise.

The Recall metric grew as the number of suggestions was raised, as seen in Fig. 8. This is because the Top-N suggestion now includes more precise POIs. Consequently, in terms of Recall, BERT-LSTM-hybrid outperformed previous methods. Asymmetric neural strategies outperform other approaches, according to the result. Compared to the other approaches, the CF and PR approaches generated the lowest accurate findings caused by the absence of clustering and disrespect for context.

The impact of the highest suggestions

As seen in Fig. 9, as the figures for suggestions rose, the precision dropped. The cause that the Highest suggestion

now contains more accurate POIs was the primary driver for this enhancement. Tourists may not be able to visit all of the suggested places due to a lack of information on personal intentions. The results show that asymmetric approaches outperformed symmetric techniques, even though the suggested technique beats the other methods in terms of precision rate. Furthermore, compared to the other approaches, the PR and CF approaches generated the lowest accurate findings caused by the absence of clustering and disrespect for context.

For numerous suggestions based on F-Score measures, BERT-LSTM-hybrid outperformed other techniques, as shown in Fig. 10. The suggested technique beats previous alternatives regarding cold start and data sparsity, as seen in this figure. By combining information in user profiles with an asymmetric LSTM schema technique to estimate the desired person's nearest neighbors, BERT-LSTM-hybrid was able to produce improved outcomes. Furthermore, user reviews and demographic data on users might help forecast user preferences for future visits, alleviating the cold start problem. Instead of depending on a single site to discover POIs, a clustering approach can assist ease the data sparsity problem. This framework suggestion may also be customized by integrating BERT with the Top-N POIs.

The distance between the expected and actual rating is calculated using the RMSE measure. This measure is frequently used in recommender systems to quantify the diversity between an item's real and projected ratings. Non-context-aware procedures, in general, have a larger mistake rate than context-aware approaches, as seen in Fig. 11. The suggested method beat other contextual approaches proposed in earlier research and had a lower error rate than non-contextual approaches. BERT-LSTM-hybrid was able to handle the cold start issue better than other methods due to the inclusion of user feedback and demographic data.
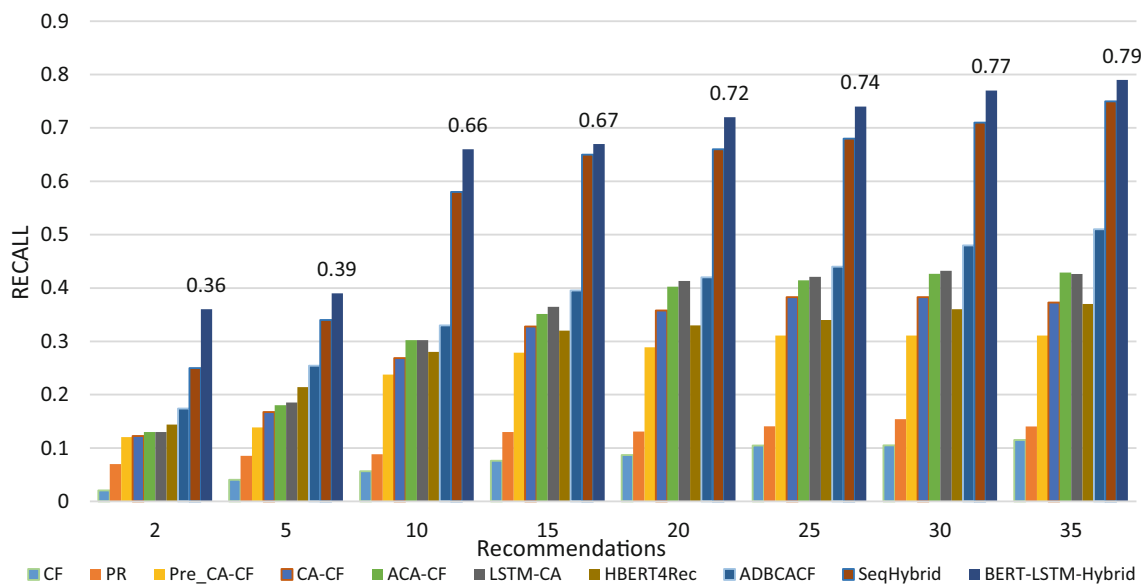
Fig. 8 The list of recommendations is used to compare BERT-LSTM-hybrid to others concerning the recall metric (Yelp)
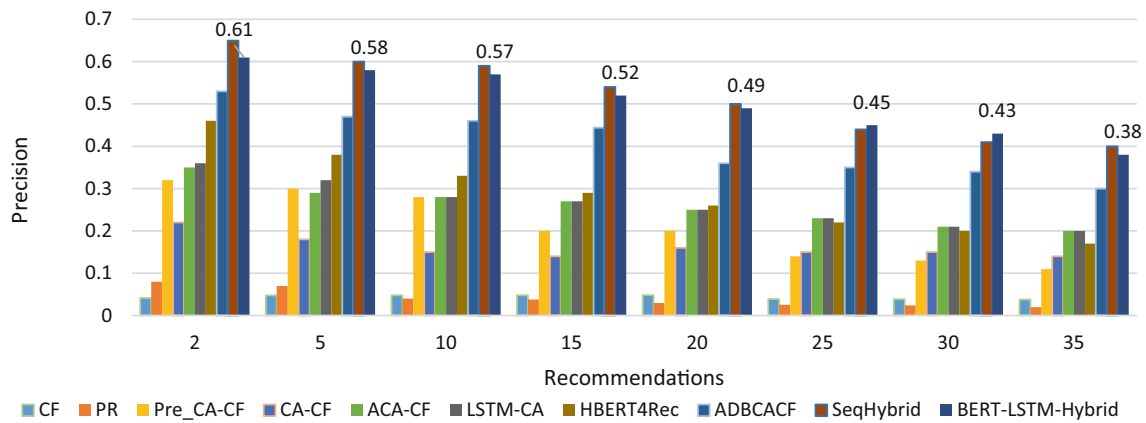

Fig. 9 The list of recommendations is used to compare BERT-LSTM-hybrid to others concerning the precision metric (Yelp)
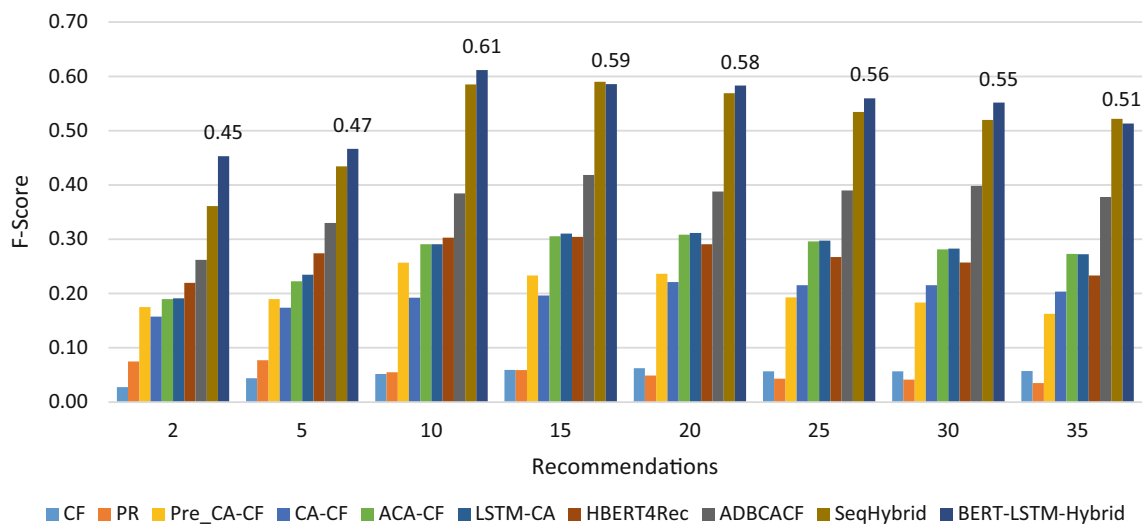

Fig. 10 The list of recommendations is used to compare BERT-LSTM-hybrid to others concerning the F-Score metric (Yelp)
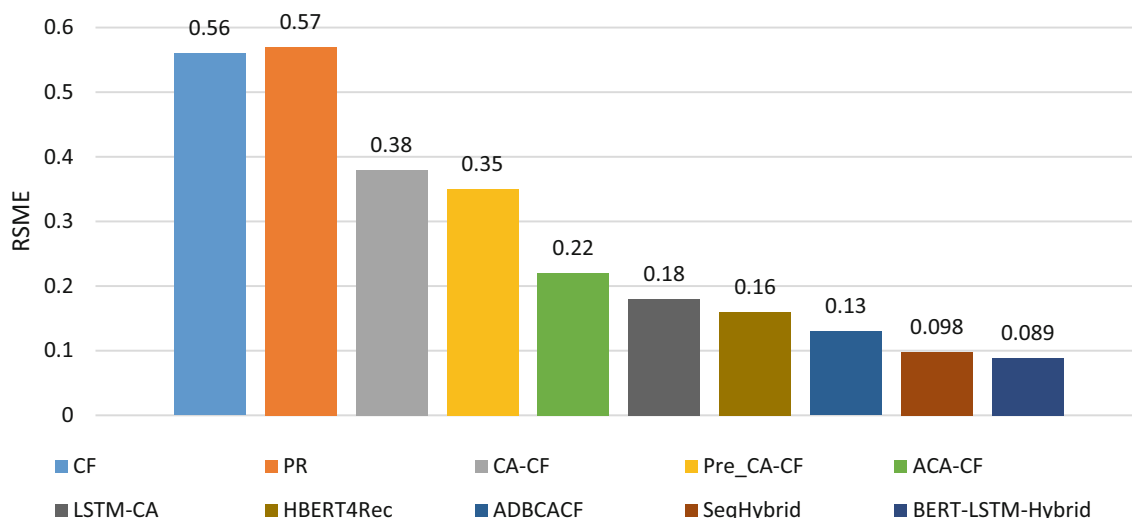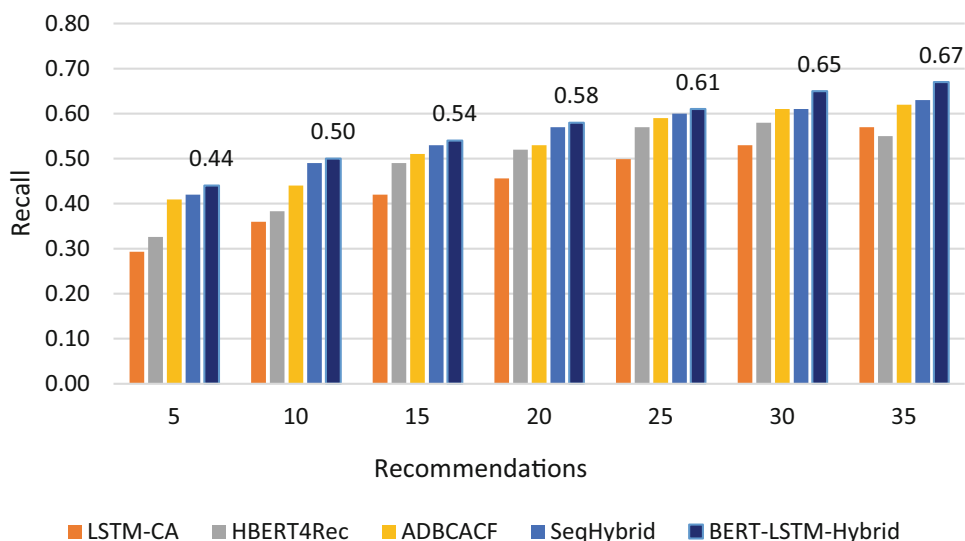
Fig. 11 BERT-LSTM-hybrid is compared to others according to the RMSE measure (Yelp)

Fig. 12 BERT-LSTM-hybrid is compared to others according to the recall measure based on the number of recommendations (tripadvisor)

Evaluation of BERT-LSTM-hybrid with the tripadvisor

Throughout this work, the tripadvisor data source was utilized as another dataset to assess BERT-LSTM-hybrid. The outcomes of evaluations, which are based on recall and precision measures, are shown in Figures 12 and 13. BERT-LSTM-hybrid yielded better results due to incorporating demographic data, contextual data, and an asymmetric schema. Compared to prior alternatives, the proposed solution proved very successful in addressing the cold start issue. When these two datasets were compared, it was discovered that tripadvisor had poorer precision and recall than Yelp, probably because Tripadvisor had fewer review and demographic characteristics. The volume of the data in this data source, as well as the number of neighbors, has an impact on the outcomes.

Evaluation of BERT-LSTM-hybrid by nDCG metric

In sequential approaches, the nDCG measure is utilized to quantify the ranking efficiency of expected suggestions. The NDCG measure highlights the rating proficiency of the sequence suggestion strategy in Fig. 14. These results showed that BERT-LSTM-hybrid projected more suitable recommendations when compared to previous approaches.

Implementation scenarios

The proposed model operates in two different scenarios: the "pertaining BERT" scenario and the "fine-tuning BERT" scenario. In the "pertaining BERT" scenario, the model utilizes the pre-trained BERT model developed by Google, which has been trained on a large corpus of general text data. This sce-
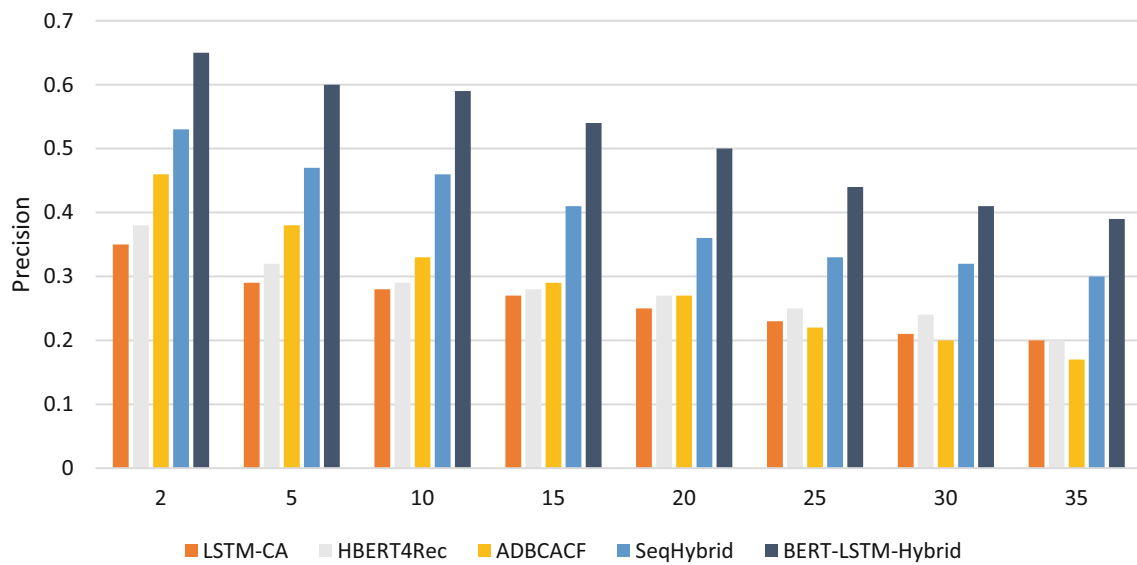
Fig. 13 BERT-LSTM-hybrid is compared to others according to the precision measure base on the number of recommendations (tripadvisor)
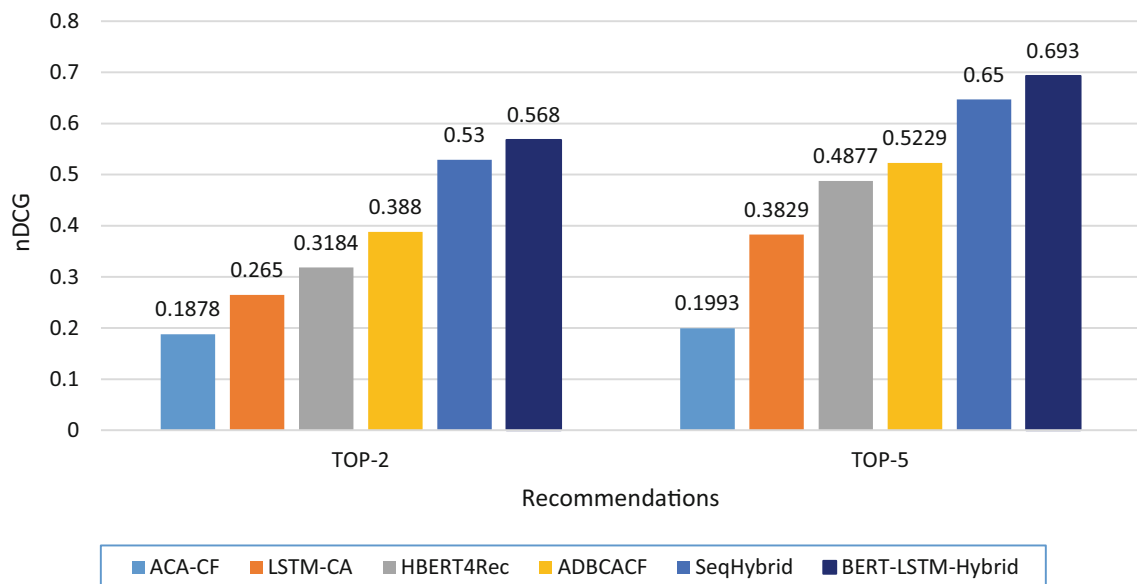

Fig. 14 BERT-LSTM-hybrid is compared to others according to the nDCG measure

nario involves using BERT as a feature extractor or encoder to obtain contextualized word representations. These representations are then used for the task at hand without further fine-tuning. In the "fine-tuning BERT" scenario, the model takes the pre-trained BERT model and further trains it on our specific task. This fine-tuning process involves updating the weights of the BERT model based on task-specific data to make it more suitable for the target task.

The choice between the two scenarios depends on the available resources, the size and specificity of the task-specific data, and the desired performance. If the task-specific data is limited, such as our dataset, fine-tuning BERT may not achieve better results compared to the pre-trained model.

However, if there is sufficient task-specific data and the computational resources are available, fine-tuning BERT can leverage the general knowledge learned from pre-training and adapt it to the specific task, potentially leading to better outcomes.

In our study, we utilized the nDCG metric to assess the ranking quality of sequence POI recommendations. The results from this metric clearly indicate that the pre-trained BERT model outperforms our fine-tuned BERT model, as depicted in Fig. 15.

In the two operation scenarios considered, the BERT-LSTM-hybrid model demonstrates notable advancements when utilizing the pre-trained BERT model, as opposed to

Fig. 15 Comparison of BERT-LSTM-hybrid in two scenarios based on nDCG measure
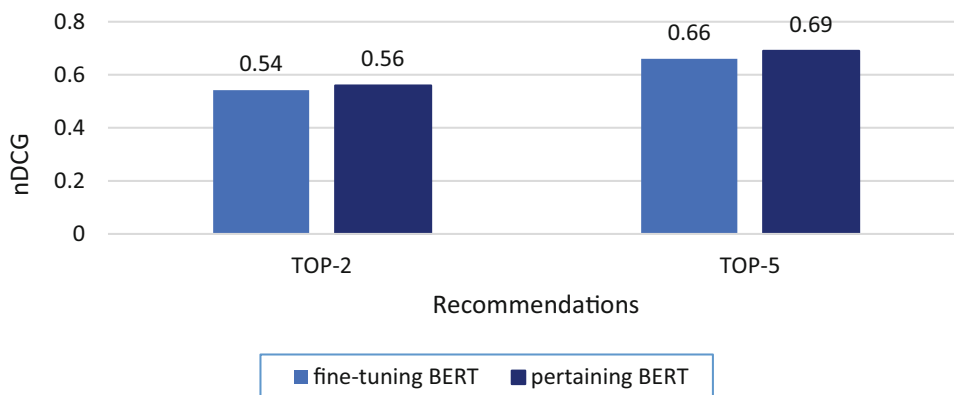
fine-tuning the BERT model. Due to the constraints of limited available resources and a relatively small dataset in comparison to Google's resources, fine-tuning the BERT model is not suitable and does not yield the desired performance. To be more precise, when the task-specific data is limited, such as in our dataset, fine-tuning BERT may not result in improved performance compared to utilizing the pre-trained model.

Impact of layer count on fine-tuning BERT performance

It's important to strike a balance when choosing the number of layers for fine-tuning BERT. Too few layers may lead to an underutilization of the model's capacity, while an excessive number of layers may introduce unnecessary complexity and computational overhead. Conducting experiments and evaluating the performance of BERT with different layer counts can help determine the optimal configuration for a specific task.

In Fig. 16, we observe the varying effects of different layer counts on the performance of our proposed model during the fine-tuning process of BERT, specifically in terms of *F*-Score metrics. The results highlight that the number of layers indeed plays a significant role. When the model has a lower number of layers, it struggles to effectively learn dependency information and surrounding context. Conversely, an excessive number of layers can lead to overfitting and the generation

of redundant information. As depicted in Fig. 16, the optimal performance for our fine-tuned BERT model is achieved with six layers. This finding demonstrates that careful consideration and experimentation with layer count can lead to improved results in the fine-tuning process.

Case study

As a case study, this paper investigates a visitor who visits a city. Once a recommendation is requested, this tourist's present location is dynamically determined using information from their cell phone. Next, potential locations consist of the city's AOIs and POIs. Since multiple POIs exist in each city and a tourist is unable to visit all, plus, as it is assumed that the tourist does not possess a sufficient history of visiting POIs in a given city (cold start problem), their nearest neighbors are recognized based on their current contexts, tourist reviews, and preferences. Finally, visitors are advised to list the top N POI routes based on their activities and information about their nearby neighbors (Fig. 17). In this investigation, we consider all data received from tourist visits and tourist reviews and present dynamic preferences based on the time and place of the targeted tourist. Therefore, with a cold start and scarce information scenarios, we may utilize all existing data to provide accurate suggestions without the direct participation of visitors.

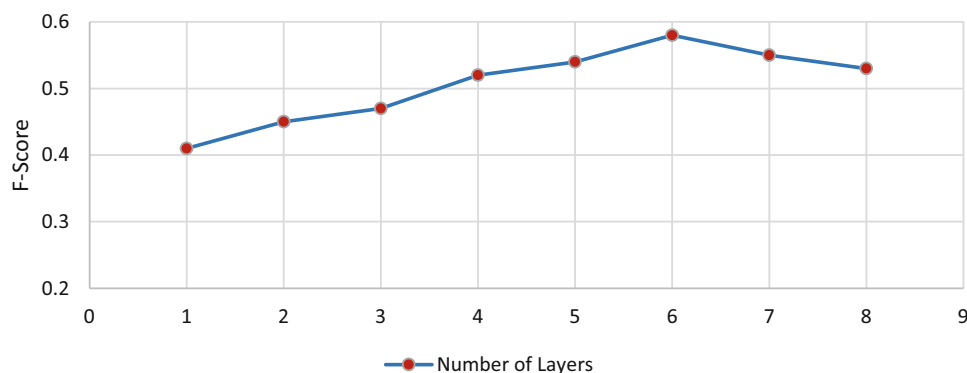**Fig. 16** The impact of the number of layers according to *F*-Score

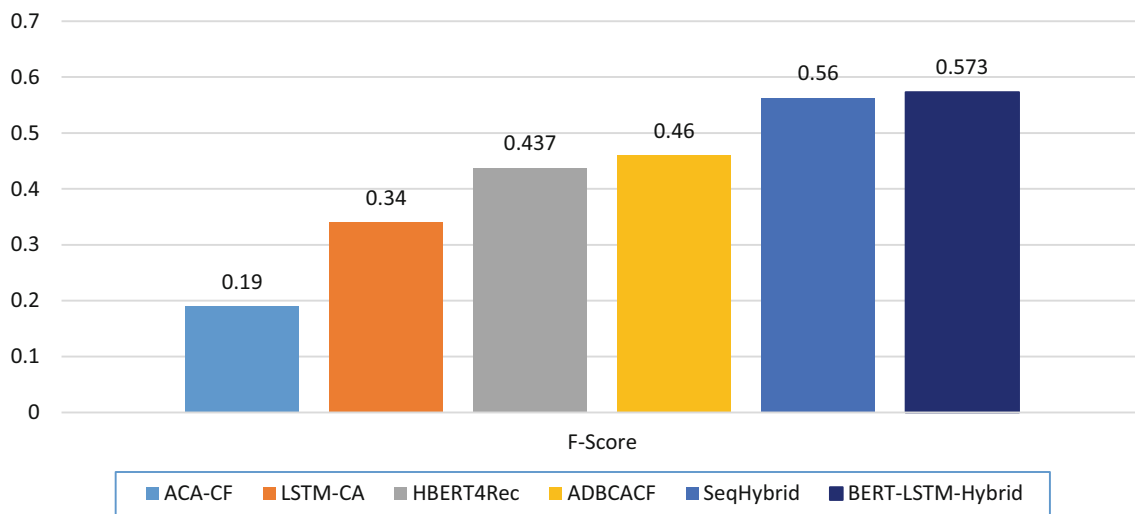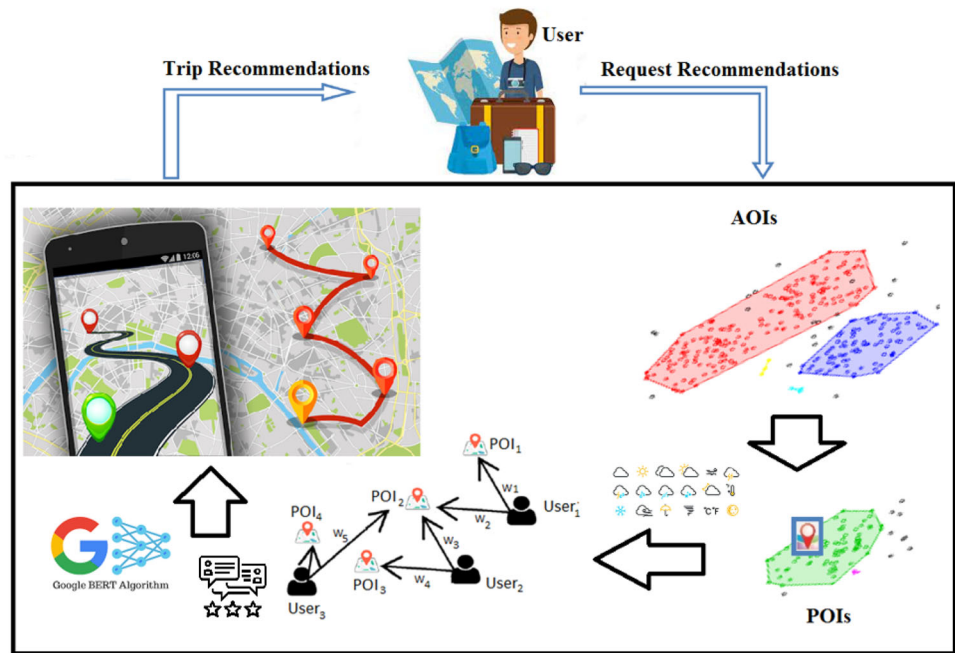**Fig. 17** Example trip recommended by BERT-LSTM-hybrid

Fig. 18 Results for performances of some methods

During the case study, we conducted further trials in the London metropolitan region to identify routes for active visitors. Similar neighbors are identified according to their current settings, reviews, histories, and qualities once an active tourist seeks a recommendation. The top N recommendations are then picked and presented. Performance evaluations of this stage are shown in Fig. 18. Based on the results, the recommended technique exhibits the highest F-Score based on an asymmetric neural CF scheme. The approaches that varied somewhat from ours ranked second. Nevertheless, the distinction between the two methods is modest, as tourists in this city have tastes and knowledge that are more similar.

Conclusions and future work

This article presents a unique neural context-aware RS for personalized tourist destinations using contextual and demographic data, user reviews, and geo-tagged social network photos. The researchers used innovative asymmetric schema, context-aware filtering, neural network, and BERT algorithms to construct a hybrid RS. Because most recommender systems rely on weak data, demographic data, and user reviews were explored to manage the cold start problem.

The recommended technique outperformed prior approaches due to the integration of contextual information and the fact that it was employed for both contextual

pre-filtering and contextual modeling. When producing tourist suggestions, it was determined that every user's context is critical. The suggested technique's personalization refers to how it uses the user's choices. Furthermore, the proposed technique improved from using DBSCAN clustering at two levels to detect POIs in any area, making clustering detection easier and more complicated. The TF-IDF approach was used to assess context similarity.

Additionally, the BERT approach is applied, which outperformed the other approaches due to the BERT algorithm's sequential movement pattern. Two data sources were examined to evaluate the efficacy of this technique (Yelp and tripadvisor). According to the comparison results, the recommended strategy can offer more precise locations than other ways. The proposed technique beat all current recommendation systems in terms of discovering users more similar to the present tourist and showed superior outcomes while coping with data sparsity and cold start difficulties.

As a result of incorporating tourists' opinions regarding their trips, the results of this method improved significantly. In other words, the results were improved when computing review similarity based on the LSTM representation of reviews. The majority of tourists prefer to elaborate on their visit and travel experience. These descriptions are available as user reviews greatly assist our model in locating comparable users. Thus, the CF model was improved.

Furthermore, because customers are more likely to accept an RS that makes ideas based on their likes and interests, this article used the rating median to indicate a unique enhanced ACF technique based on user preferences. The study's journey sequences would help travelers plan their vacations and make them more convenient. This method increases the interactivity of the tourist recommender system by deriving trip patterns from visitor behavior. This technology increases user engagement with online trip RS by recognizing and tailoring journey patterns based on users' travel behavior. Tourist behavior patterns can give insight into visitors' intentions and desires by anticipating users' future interests and activities based on recent behaviors.

In terms of language comprehension, BERT has proven to be extremely effective. As a result, POI hybrid recommendations have been made using the BERT in this paper. BERT architecture is used in the proposed model, which includes an item encoder for POI items and a recommendation and preference prediction task. Using the proposed technique, users can be advised of POIs based on their next POIs of interest and trips, and a prediction is made for each of them. An extensive set of experiments on two real-world datasets have demonstrated the superiority of this model over established benchmarks.

It is noteworthy that, the approach of using BERT and LSTM for sequential neural recommendation systems on social media posts has its limitations and weak points. One

limitation of this approach is the amount of training data required to effectively train the model. The performance of neural networks is highly dependent on the amount and quality of data available for training. If the training data is insufficient or biased, the model's performance may suffer. Additionally, the model may overfit the training data, resulting in poor generalization to new data. Another limitation is the complexity of the model. Neural networks can be highly complex and require a significant amount of computational resources to train and deploy. This can be a challenge, particularly for smaller organizations or those with limited resources.

Future work

There are multiple potential avenues for extending the approach presented in this study. One way to extend the approach is to use relationships between users and the impact of companions to increase accuracy further. Using the Transform layer to transform the tourist-POI joint feature map obtained from CNN into an approximation of their joint review, which could provide intelligence into the user's experience with the POI throughout train travel and evaluation, is another method for improving the model's performance. These factors may increase the effectiveness of the tourist recommender method because they can interact with diverse tourist preferences.

Moreover, future trends in the development of neural recommendation systems that utilize deep learning models and contextual information from social media posts are likely to focus on several areas, including explainability, privacy, and personalization. As users continue to demand more transparency and control over their data, future recommendation systems are likely to incorporate methods for explaining the recommendations made by the model and protecting user privacy.

In the context of the Internet of Things (IoT) technology development, the integration of recommendation systems with IoT devices and sensors can provide additional contextual information that can improve the accuracy and relevance of recommendations. For example, a recommendation system that utilizes data from a user's wearable device or smart home sensors can make more accurate recommendations based on the user's behavior and context.

However, the development of such recommendation systems requires significant computational resources and memory due to the complex deep learning models used, such as BERT and LSTM. Therefore, the model would benefit from being deployed on a high-performance computing infrastructure, such as a cloud computing platform or a cluster of graphics processing units (GPUs). Additionally, the model may require data pre-processing and normalization to handle the variety of social media data formats and to ensure

consistency in the data representation. The necessary transfer and network configurations would depend on the specific deployment environment and the requirements of the recommendation system. Overall, the development of neural recommendation systems that utilize deep learning and social media data has great potential to provide personalized and relevant recommendations to users, and will likely continue to be an area of active research and development in the future.

Declarations

Conflict of interest None.

References

1. Noorian A, Ravanmehr R, Harounabadi A, Nouri F (2020) Trust-based tourism recommendation system using context-aware clustering. Tour Manag Stud 15:309–344
2. Bathla G, Singh P, Kumar S et al (2021) Recop: fine-grained opinions and sentiments-based recommender system for industry 50. Soft Comput. https://doi.org/10.1007/S00500-021-06590-8
3. Adomavicius G, Bauman K, Tuzhilin A, Unger M (2022) Context-aware recommender systems: from foundations to recent developments. In: Ricci F, Rokach L, Shapira B (eds) Recommender systems handbook. Springer, New York, pp 211–250
4. Al-Shamri MYH (2022) Similarity modifiers for enhancing the recommender system performance. Appl Intell 52:8534–8550
5. Noorian Avval AA, Harounabadi A (2023) A hybrid recommender system using topic modeling and prefixspan algorithm in social media. Complex Intell Syst. https://doi.org/10.1007/s40747-022-00958-5
6. Deldjoo Y, Schedl M, Hidasi B et al (2022) Multimedia recommender systems: algorithms and challenges. Recomm Syst Handb. https://doi.org/10.1007/978-1-0716-2197-4_25
7. Duan R, Jiang C, Jain HK (2022) Combining review-based collaborative filtering and matrix factorization: a solution to rating's sparsity problem. Decis Support Syst. https://doi.org/10.1016/j.dss.2022.113748
8. Wang X, Fukumoto F, Li J et al (2022) STaTRL: spatial-temporal and text representation learning for POI recommendation. Appl Intell. https://doi.org/10.1007/s10489-022-03858-w
9. Sarkar JL, Majumder A (2022) gTour: multiple itinerary recommendation engine for group of tourists. Expert Syst Appl. https://doi.org/10.1016/j.eswa.2021.116190
10. Noorian A (2024) A BERT-based sequential POI recommender system in social media. Comput Stand Interfaces 87:103766. https://doi.org/10.1016/j.csi.2023.103766
11. Hamid RA, Albahri AS, Alwan JK et al (2021) How smart is e-tourism? A systematic review of smart tourism recommendation system applying data management. Comput Sci Rev 39:100337
12. Wu J, Hu R, Li D et al (2022) Where have you been: dual spatiotemporal-aware user mobility modeling for missing check-in POI identification. Inf Process Manag 59:103030
13. Ye X, Liu D (2022) A cost-sensitive temporal-spatial three-way recommendation with multi-granularity decision. Inf Sci (Ny) 589:670–689. https://doi.org/10.1016/j.ins.2021.12.105
14. Devlin J, Chang MW, Lee K, Toutanova K (2019) BERT: Pre-training of deep bidirectional transformers for language understanding. In: NAACL HLT 2019 - Proceedings of the 2019 conference of the North American chapter of the Association for Computational Linguistics: human language technologies, vol 1, pp 4171–4186
15. Ray B, Garain A, Sarkar R (2021) An ensemble-based hotel recommender system using sentiment analysis and aspect categorization of hotel reviews. Appl Soft Comput 98:106935. https://doi.org/10.1016/j.asoc.2020.106935
16. Feng J, Xia Z, Feng X, Peng J (2021) RBPR: a hybrid model for the new user cold start problem in recommender systems. Knowl Based Syst 214:106732
17. Heidari N, Moradi P, Koochari A (2022) An attention-based deep learning method for solving the cold-start and sparsity issues of recommender systems. Knowl Based Syst 256:109835
18. Huang S, Wu X, Wu X, Wang K (2023) Sentiment analysis algorithm using contrastive learning and adversarial training for POI recommendation. Soc Netw Anal Min 13:1–14. https://doi.org/10.1007/S13278-023-01076-X/TABLES/9
19. Noorian A, Harounabadi A, Ravanmehr R (2022) A novel sequence-aware personalized recommendation system based on multidimensional information. Expert Syst Appl 202:117079
20. Ahmadian M, Ahmadi M, Ahmadian S (2022) A reliable deep representation learning to improve trust-aware recommendation systems. Expert Syst Appl. https://doi.org/10.1016/j.eswa.2022.116697
21. Wahab OA, Rjoub G, Bentahar J, Cohen R (2022) Federated against the cold: a trust-based federated learning approach to counter the cold start problem in recommendation systems. Inf Sci 601:189–206
22. Islam MA, Mohammad MM, Sarathi Das SS, Ali ME (2022) A survey on deep learning based Point-of-Interest (POI) recommendations. Neurocomputing 472:306–325. https://doi.org/10.1016/j.neucom.2021.05.114
23. Zheng Y, Wang D (2022) A survey of recommender systems with multi-objective optimization. Neurocomputing 474:141–153. https://doi.org/10.1016/j.neucom.2021.11.041
24. Tan KS, Lim KM, Lee CP, Kwek LC (2022) Bidirectional long short-term memory with temporal dense sampling for human action recognition. Expert Syst Appl. https://doi.org/10.1016/j.eswa.2022.118484
25. Li C, Xu L, Yan M, Lei Y (2020) TagDC: a tag recommendation method for software information sites with a combination of deep learning and collaborative filtering. J Syst Softw 170:110783. https://doi.org/10.1016/j.jss.2020.110783
26. Wang K, Wang X, Lu X (2021) POI recommendation method using LSTM-attention in LBSN considering privacy protection. Complex Intell Syst. https://doi.org/10.1007/S40747-021-00440-8
27. Sun K, Qian T, Chen T et al (2020) Where to go next: modeling long- and short-term user preferences for point-of-interest recommendation. AAAI 2020—34th AAAI conference on artificial intelligence, New York, USA, pp 214–221. https://doi.org/10.1609/AAAI.V34I01.5353

28. Zhao K, Zhang Y, Yin H et al (2020) Discovering subsequence patterns for next POI recommendation. IJCAI Int Jt Conf Artif Intell 2021:3216–3222. https://doi.org/10.24963/ijcai.2020/445

29. Bai Z, Li Y, Woźniak M et al (2021) DecomVQANet: decomposing visual question answering deep network via tensor decomposition and regression. Pattern Recognit. https://doi.org/10.1016/j.patcog.2020.107538

30. Zhuang Z, Tao H, Chen Y et al (2022) An optimal iterative learning control approach for linear systems with nonuniform trial lengths under input constraints. IEEE Trans Syst Man Cybern Syst. https://doi.org/10.1109/TSMC.2022.3225381

31. Tao H, Li X, Paszke W et al (2021) Robust PD-type iterative learning control for discrete systems with multiple time-delays subjected to polytopic uncertainty and restricted frequency-domain. Multidimens Syst Signal Process 32:671–692. https://doi.org/10.1007/S11045-020-00754-9

32. Zhou C, Tao H, Chen Y et al (2022) Robust point-to-point iterative learning control for constrained systems: a minimum energy approach. Int J Robust Nonlinear Control 32:10139–10161. https://doi.org/10.1002/RNC.6354

33. Choudhary M, Chouhan SS, Pilli ES, Vipparthi SK (2021) BerConvoNet: a deep learning framework for fake news classification. Appl Soft Comput 110:107614

34. Zhang Y, Yin G, Dong H, Zhang L (2022) Attention-based frequency-aware multi-scale network for sequential recommendation. Appl Soft Comput 127:109349. https://doi.org/10.1016/j.asoc.2022.109349

35. Fan J, Yu X, Wang Z et al (2022) AUBRec: adaptive augmented self-attention via user behaviors for sequential recommendation. Neural Comput Appl 34:21715–21728

36. Qiu R, Huang Z, Yin H, Wang Z (2022) Contrastive learning for representation degeneration problem in sequential recommendation. In: Proceedings of the fifteenth ACM international conference on web search and data mining. ACM, New York

37. Islam S, Bhattacharya S (2022) Ar-bert: aspect-relation enhanced aspect-level sentiment classification with multi-modal explanations. In: Proceedings of the ACM web conference 2022, Lyon, France, pp 987–998

38. Sun F, Liu J, Wu J, et al (2019) Bert4rec: sequential recommendation with bidirectional encoder representations from transformer. In: Proceedings of the 28th ACM international conference on information and knowledge management, pp 1441–1450. https://doi.org/10.1145/3357384.3357895

39. Yang N, Jo J, Jeon M et al (2022) Semantic and explainable research-related recommendation system based on semi-supervised methodology using BERT and LDA models. Expert Syst Appl. https://doi.org/10.1016/j.eswa.2021.116209

40. Seol JJ, Ko Y, Lee S-G (2022) Exploiting session information in BERT-based session-aware sequential recommendation. In: Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval. ACM, New York

41. Zhuang Y, Kim J (2021) A BERT-based multi-criteria recommender system for hotel promotion management. Sustain 13:8039. https://doi.org/10.3390/SU13148039

42. Wang Z, Zeng J, Wen J et al (2023) Point-of-interest recommendation using deep semantic model. Expert Syst Appl. https://doi.org/10.1016/J.ESWA.2023.120727

43. Thaipisutikul T, Chen YN (2023) An improved deep sequential model for context-aware POI recommendation. Multimed Tools Appl. https://doi.org/10.1007/S11042-023-15540-5/METRICS

44. Xu Y, Wu Y, Gao H et al (2021) Collaborative APIs recommendation for artificial intelligence of things with information fusion. Future Gener Comput Syst 125:471–479

45. Thakkar A, Chaudhari K (2021) A comprehensive survey on deep neural networks for stock market: the need, challenges, and future directions. Expert Syst Appl 177:114800

46. Lauriola I, Lavelli A, Aiolli F (2022) An introduction to deep learning in natural language processing: models, techniques, and tools. Neurocomputing 470:443–456

47. Pirasteh P, Hwang D, Jung JJ (2015) Exploiting matrix factorization to asymmetric user similarities in recommendation systems. Knowl Based Syst 83:51–57. https://doi.org/10.1016/j.knosys.2015.03.006

48. Lim KH, Chan J, Leckie C, Karunasekera S (2018) Personalized trip recommendation for tourists based on user interests, points of interest visit durations and visit recency. Knowl Inf Syst 54:375–406

49. Bashir SR, Misic V (2022) BERT4Loc: BERT for Location—POI recommender system. Future Internet 15:213

50. Fan J, Gao X, Wang T et al (2021) Research and application of automated search engine based on machine learning. In: 2021 international conference on high performance big data and intelligent systems. IEEE

51. Wilkinson PF (2015) User personality and the new user problem in a context-aware point of interest recommender system. Ann Tour Res 28:1070–1072. https://doi.org/10.1016/S0160-7383(01)00012-3

52. Gomes L, Torres RS (2022) Software MC-I and, 2023 undefined BERT-and TF-IDF-based feature extraction for long-lived bug prediction in FLOSS: a comparative study. Elsevier, New York

53. Catelli R, Fujita H, De PG et al (2022) Deceptive reviews and sentiment polarity: effective link by exploiting BERT. Elsevier, New York

54. Nozza D, Bianchi F, Hovy D (2020) What the [MASK]? Making sense of language-specific BERT models. arXiv preprint arXiv:2003.02912

55. Haruna K, Ismail MA, Suhendroyono S et al (2017) Context-aware recommender system: a review of recent developmental process and future research direction. Appl Sci 7:1–25. https://doi.org/10.3390/app7121211

56. Bobadilla J (2018) Reliability quality measures for recommender systems. Inf Sci (Ny). https://doi.org/10.1016/j.ins.2018.02.030

57. Liu X, Yang Y, Xu Y et al (2022) Real-time POI recommendation via modeling long-and short-term user preferences. Neurocomputing 467:454–464

58. Joorabloo N, Jalili M, Ren Y (2022) Improved recommender systems by denoising ratings in highly sparse datasets through individual rating confidence. Inf Sci 601:242–254

59. Ma Y, Mao J, Ba Z, Li G (2020) Location recommendation by combining geographical, categorical, and social preferences with location popularity. Inf Process Manag 57:102251

60. Kefalas P, Manolopoulos Y (2017) A time-aware spatio-textual recommender system. Expert Syst Appl 78:396–406. https://doi.org/10.1016/j.eswa.2017.01.060

61. Linda S, Bharadwaj KK (2019) A genetic algorithm approach to context-aware recommendations based on spatio-temporal aspects. In: Integrated intelligent computing, communication and security. Springer, pp 59–70

62. Wang D, Xu D, Yu D, Xu G (2021) Time-aware sequence model for next-item recommendation. Appl Intell 51:906–920

63. Yang N, Jo J, Jeon M et al (2022) Semantic and explainable research-related recommendation system based on semi-supervised methodology using BERT and LDA models. Expert Syst Appl 190:116209

64. Kolahkaj M, Harounabadi A, Nikravanshalmani A, Chinipardaz R (2020) A hybrid context-aware approach for e-tourism package recommendation based on asymmetric similarity measurement and sequential pattern mining. Electron Commer Res Appl 42:100978. https://doi.org/10.1016/j.elerap.2020.100978

Springer