IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 8 May 2024, accepted 12 July 2024, date of publication 19 July 2024, date of current version 11 September 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3430987

SURVEY

A Review of User Profiling Based on Social Networks

WENBO WU[1,2], MASITAH GHAZALI[3], AND SHARIN HAZLIN HUSPI[1]
[1]Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru 81310, Malaysia
[2]School of Computer Information, Minnan Science and Technology College, Quanzhou 362332, China
[3]Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia, Kuala Lumpur 54100, Malaysia

Corresponding author: Wenbo Wu (wuwenbo971@gmail.com)

This work was supported in part by the Department of Education, Fujian, China, under Grant FBJG20210123 and Grant JAT231173.

ABSTRACT The rapid development of the internet and smartphones has enabled people to access numerous information systems and large volumes of data. User profiling technology can meet the dual challenge of analyzing user characteristics, interests, or preferences and recommending corresponding resources. Nevertheless, the insufficiency and isolation of data in traditional information systems limit the effect of user profiling, and social networks can compensate for this deficiency. With massive quantities of data in text, images, videos, and relationships in social networks, user profiling can achieve highly accurate analytical results. This review comprehensively discusses user profiling for social networks, defines its criteria, and expounds on the entire process, from data collection to the profiling model and performance evaluation. It includes various models with advantages and disadvantages and corresponding application scenarios. Additionally, considering that technology serves humans, this review provides users with multiple applications in the industry for user profiling based on social networks. Furthermore, it discusses the ethical and legal issues associated with user profiling. Finally, this review highlights possible future research directions in this field. Overall, this review can help researchers enhance their understanding of the current state of research in the field of user profiling and gain ideas for further study.

INDEX TERMS Recommendation system, social network, user modeling, user profiling.

I. INTRODUCTION

The rapid development of information and communication technology has resulted in an urgent need for personalized information that can be fulfilled by predicting user profiles such as demographic data, personalities, and preferences, which can help match appropriate resources to particular users [1], [2]. To achieve this goal, user profiling can be used to predict different user profiles, such as gender, age, personality, and interest, based on user posts, behaviors, and relationships [3]. User profiling has been widely used in various scenarios including session-based recommendations [4], online learning [5], location analysis [6], community question-answering systems [7], contribution adaptation [8], and mental state recognition [9].

The data sources used in user profiling have evolved over time. Early user profiling used the operator records of information systems. For example, Hong et al. [10] used job seeker information in a recruitment system to provide job recommendations. Jomsri [11] proposed obtaining data from the borrowing records of a library-lending system. These data were obtained only from isolated information systems, and could not be shared with others. With the rapid development of smartphones and the internet, internet-based information systems now offer abundant data, including Wi-Fi device logs [12], user–mobile application interactions [13], and application server logs [14]. Because such data often come directly from users, the data acquisition approach has become increasingly challenging because users are less willing to share private information publicly [15].

Social networks are advantageous as user profiling data sources. Today, users frequently post text, photos, and videos on social networks, such as Twitter, Facebook, Instagram,

The associate editor coordinating the review of this manuscript and approving it for publication was Filbert Juwono.

and TikTok. Simultaneously, the users establish different relationships. This abundant data provides excellent support for user profiling [16]. Biswas et al. [9] predicted user personalities based on Twitter content and text. Stefanovič and Ramanauskaitė [17] used users' Instagram photos to make travel recommendations. Gomez et al. [18] identified the age and sex of users by using multilingual datasets collected from Twitter and Pinterest. Nagar et al. [19] detected hate speech based on text in social networks. Dehshibi et al. [20] predicted five basic prototypical user requirements using Instagram images.

Subsequent research will benefit significantly from summarizing the research results and elaborating on the technologies used in user profiling based on social networks. Several reviews have already been published in this field. Kanwal et al. [21] focused on text-based recommendations, including news research articles, e-books, and personal blogs, in which text information can be used for user analysis and targeted recommendations. Eke et al. [15] reviewed user profiling research application scenarios, defined user profiling classifications, and explained standard models and techniques. Piao and Breslin [22] stated user profiling based on microblogging, which can be used to infer user interests. However, there are research gaps in the above studies. Some studies should narrow their scope and focus on specific areas like social networks. Some research data sources are limited to text and need to consider richer data sources such as images, videos, location, and relationships. Most studies require a detailed description of the model algorithm, which could be more conducive to understanding and applying the model.

Based on the above, the research questions of this review are as follows:

- What are the findings of the study, how is the industry utilizing these findings, and where might future research in social-network-based user profiling go?
- Which technologies have been used at each stage of user profiling, and what is their specific operation mechanism?
- How can the effectiveness of a social-network-based user profiling model be assessed, and what are the ethical and legal aspects of user profiling based on social networks?

The research objectives correspond to the above research questions and are as follows:

- To obtain research works, explore industrial applications, and discuss potential research directions in social network-based user profiling to lay the groundwork for future research.
- To describe which technologies are currently in use and how they function at each step of the user-profiling model.
- To compile evaluation indicators using the social network-based user profiling approach, and discuss the ethical and legal issues of user profiling.

Compared to previous research, this review focuses on user profiling based on social networks, involves various data sources, explains each process phase, and discusses potential research directions. So, this review makes the following contributions:

- *We discuss definitions and criteria for user profiling and social networks.*
- *We highlight the processing stages of user profiling based on social networks in typical application scenarios and elaborate on each algorithm in the computing model.*
- *We introduce various industrial applications of user profiling based on social networks and discuss the ethical and legal issues from a human-centered perspective.*
- *We discuss the potential research direction in the field of user profiling based on social networks for subsequent research.*

The remainder of this paper is organized as follows: Section II states the methodology of this review. Section III presents the study background. Section IV describes the primary processing phases: data collection, preprocessing, feature extraction, modeling technologies, and performance evaluation. Section V displays the industrial applications. Section VI discusses the ethical and legal aspects of this study area. Section VII provides the future research directions. Finally, Section VIII concludes the paper.

II. METHODOLOGY

A. PAPER SEARCH

This study conducts a literature review by searching well-known databases, such as Web of Science, Scopus, ACM, IEEE, Wiley, ScienceDirect, SpringerLink, and Google Scholar, using the following keywords and their synonyms: *"user profiling," "user modeling," "social network," and "recommendation system."* The following criteria were used for the selection of papers:

- We mainly consider publications within the last ten years.
- We considered only papers published in English.
- We considered only those studies whose research area was user profiling.
- We selected only those studies whose primary data sources came from social networks.
- We also selected studies relevant to the research question stated earlier.

After screening based on the above criteria, 168 papers related to this study were selected. Table 1 shows the collected papers, including journal articles, conference proceedings, book sections, and review articles, along with their respective numbers.

Fig. 1 presents a histogram of the number of papers published annually. According to the data, out of all the papers, 85.7% have been published in the past 7 years. The year 2023 had the highest number of published papers, with 26 papers, closely followed by 2022 and 2021, with 25 and

TABLE 1. Types and number of papers after searching and screening.

Paper Types	Number
Total	168
Journal article	116
Conference proceedings	49
Book section	3
Article	151
Review article	17

24 papers, respectively. Several studies have been published since 2015, while before 2015, only seven papers had been published.

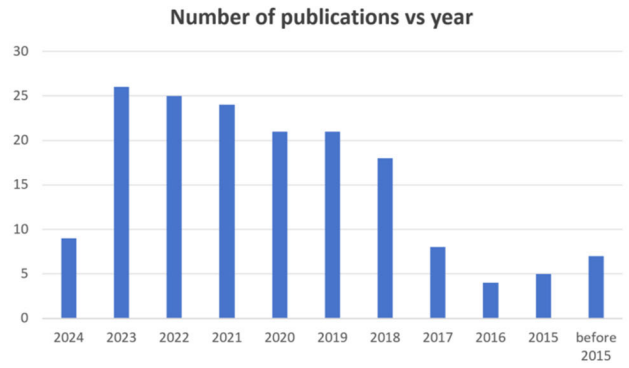

FIGURE 1. Number of papers published yearly.

Fig. 2 presents a histogram that indicates the number of papers published by each publisher. According to the histogram, IEEE published the most papers, with 38 papers, followed by Springer, ACM, and Elsevier, which published 37, 19, and 15 papers, respectively. The top nine publishers were responsible for most of the published papers, accounting for 78%. However, the remaining publishers accounted for only 22% of the total number of papers.

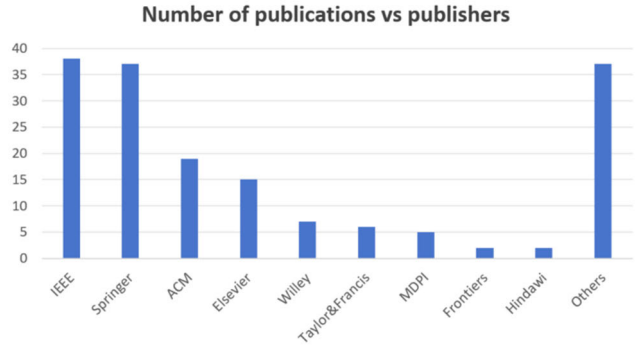

FIGURE 2. Number of papers published by each publisher.

B. DATA COLLECTION AND ORGANIZATION

After obtaining the relevant papers, we conducted a thorough analysis and summarized some particular contents, which are further discussed in subsequent sections.

- User profiling is widely used in many industries and has significant value for humans.
- User profiling studies utilize various models and techniques, covering data collection, data preprocessing, feature extraction, and user models. Each model has its advantages and disadvantages.
- The models used in the studies require specific evaluation methods and indicators to evaluate performance.
- Ethical and legal issues need to be faced in the studies.
- There are potential future research directions in the studies.

III. BACKGROUNDS

Several researchers have defined the concept of user profiles. Ouaftouh et al. [23] defined a user profile as a series of structured data that describe the interaction context between a user and a particular system. A user profile can represent the preferences or interests of a single user or group of users, including demographic statistics such as name, gender, age, interests, personal keywords, domain knowledge, and preferences. Liu et al. [24] stated that user profiling can reflect user interest, preferences, and habits. A personalized ontology can provide an effective solution for user profiles, and user intent in a web search may identify a user's interest in a query in a user ontology. Alaoui et al. [25] stated that each user is personalized and looks for the appropriate and relevant information they need. User profiles offer unique information that can clarify user needs and predict their tendencies. This information can be used to understand the user to the greatest extent possible, and to serve the user better.

A user profile can be represented by a triplet (u, P, v), where u represents a user, P is an item, and v is the attribute value of the user for an item. These triplets are often stored as rating matrices $R^{m\times n}$, as listed in Table 2, where m represents the number of users, and n represents the number of items. For each user u, the value v for each item P is provided by the user or predicted from their interaction data [26]. User profiling is an approach for inferring an unknown attribute v using various methods.

TABLE 2. Mathematical representation of user profiling. (A $R^{m\times n}$ matrix represents a user profile, where u represents a user, P represents an item, and v represents a user's profile value for an item.)

	P_1	P_2	P_3	...	P_n
u_1	v_{11}	v_{12}	v_{13}	...	v_{1n}
u_2	v_{21}	v_{22}	v_{23}	...	v_{2n}
u_3	v_{31}	v_{32}	v_{33}	...	v_{3n}
...	...	...	...	...	...
u_m	v_{m1}	v_{m2}	v_{m3}	...	v_{mn}

User profiling has several categories. Based on the time dimension, it can be classified as either static or dynamic based on the time dimension. Static profiling remains unchanged or changes slowly over time, depending on factors such as sex and age. By contrast, dynamic profiling, such as preferences and interests, changes more frequently [15].

Among the dynamic user profiles, long- and short-term profiles are subcategories. Long-term profiles are extracted from a prolonged period in the user history, whereas short-term profiles are based on a short interval period [27].

According to the data acquisition, user profiling can be classified as explicit, implicit, or hybrid. Explicit data, such as user search queries and ratings from social networks, are observed more efficiently [28]. Although this method is often more direct and accurate than other methods, it is limited by people's reluctance to provide data [29]. Implicit data are not directly available, but can be inferred from a specified source. Learning from implicit data generated from large-scale user conversation histories can be effective in personalized chatbot applications [30]. Hybrid data has the advantages of both explicit and implicit methods. A hybrid approach can achieve good results in the user's reading behavior analysis [31].

User profiling can be divided into demographic, psychographic, and wellness profiles, based on profile attributes. Demographic profiles included age, sex, race, city, country, and occupation. Psychographic profiling contains individual psychological states such as behavior, interests, preferences, attitudes, and emotions. Wellness profiling includes personal health attributes such as body mass index and disease propensity [32].

Social networking services provide space for users to interact face-to-face. Internet-mediated user interactions connect various members and may contribute to creating, maintaining, and developing new social and working relationships [26]. Moreover, social networks can help collect data generated by users on the internet [33]. With the massive volumes of data, user profiling has become increasingly effective. Twitter users can keep up with the latest developments and news on various topics of interest through the accounts they follow [34]. Social networks influence education through online learning [35]. In addition to their large scale, social network data types, including text, relationships, images, sounds, and videos, are more abundant than traditional data sources [32].

The meanings of the acronyms used in this study are listed in Table 3.

TABLE 3. Acronyms and definitions.

Acronyms	Meaning
SSID	Service Set Identifier
API	Application Programming Interface
LDA	Latent Dirichlet Allocation
K-NN	K-nearest Neighbor
NB	Naïve Bayes
SVM	Support Vector Machine
MLP	Multilayer Perceptron
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
LSTM	Long Short-term Memory
GRU	Gated Recurrent Unit
ReLu	Rectified Linear Activation Function
SVD	Singular Value Decomposition
GNN	Graph Neural Network
GCN	Graph Convolutional Network
GAE	Graph Autoencoders
GAN	Generative Adversarial Networks
KG	Knowledge Graph
LBSN	Location-based Social Network

IV. PROCESS PHASES OF USER PROFILING

This section reviews the processing of the user profiling phases required to obtain user profiles. These processing phases and data mining are conducted to generate user profiles representing user attributes, interests, and preferences, thereby enabling users to make predictions and recommendations. In general, user profiling comprises five phases: data collection, preprocessing, feature extraction, modeling, and performance evaluation [15]. These phases are explained below and presented in Fig. 3.

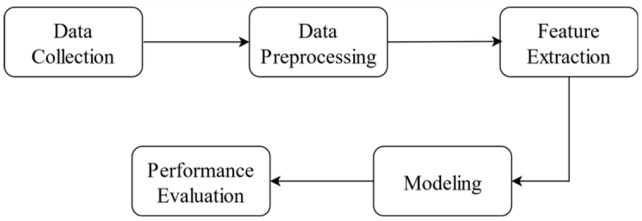

FIGURE 3. Processing phases of user profiling.

A. DATA COLLECTION

The data collection phase primarily involves collecting datasets from isolated information systems, internet-related systems, and social networks. Table 4 presents these data.

TABLE 4. Data sources.

Data Sources	Data Contents	References
Information system data	Job finders' database	[10] [39]
	Library system database	[11]
Internet-related data	Wi-Fi device logs	[12][36]
	Smartphone data	[13]
	Application server logs	[14][37]
	Text data on the internet	[21]
	User conversation histories	[30]
Social Network data	Twitter content	[9][18][19] [38]
	Instagram content	[17][20]
	Multiple platforms	[22]
	Facebook content	[34] [39]

Most of the early data were obtained from isolated information systems. The operator information left by users can be used for user profiling. Jomsri [11] proposed a model to extract data from a library lending database, whose loan records included book identity, book name, book type, author name, user identity, barcodes, user type, date borrowed, and

date returned. Hong et al. [10] proposed a career recommendation system that updates user profiles based on a recruitment system's job action and position in user history. Guo et al. [36] developed a resume-matching system that matches job seekers' qualifications with the needs of job posts.

In the internet era, people visit various websites when they perform different operations or leave behavior traces, and application server logs record this content. Hence, application server logs are an essential source of user profiling data. Mobasher [37] proposed extracting user behavior data from server data. Suchacka and Iwański [14] described sessions based on the statistical characteristics of server logs as a data source for new robot detection methods. With the widespread use of smartphones, user profiling has come to support various behavior records of mobile phone users. Amoretti et al. [13] proposed using a smartphone application to collect user data, including historical user operations and environmental data collected by the phone. Wi-Fi is often used to access the internet. Wi-Fi logs record several user behaviors and preferences, making it possible to perform user profiling. Fan et al. [12] proposed using data from Wi-Fi devices to predict user preferences. Kanwal et al. [21] listed texts that can be used for user profiling on the internet, such as news, e-books, personal blogs, and user reviews. Ma et al. [30] proposed a method to automatically learn implicit user information from large-scale user conversation history.

With the rise of social networks, people have accumulated many posts and social relations, as well as a large amount of content. Rich data on social networks provides excellent support for user profiling. Piao and Breslin [22] listed the most commonly used social networking platforms for user profiling such as Twitter, Facebook, and LinkedIn. Due to Twitter's openness to data, many studies have used Twitter content. For example, Alhozaimi and Almishari [38] collected data from tweets on Twitter in Saudi Arabia covering four popular categories: politics, economics, entertainment, and sports. Approximately 14,000 tweets were collected using Twitter API. Biswas et al. [9] used Twitter images and text as data sources to predict a user personality. Nagar et al. [19] predicted hate speech based on Twitter user data. Gomez et al. [18] used Twitter and Pinterest content to identify users' age and gender. As Facebook has numerous users and data, many studies have used it as a data source. Pereira et al. [35] designed the BROAD-RSI educational recommendation system. When a user uses Facebook, the system authorizes access to information files. The system then reads user profiles, friend lists, favorite pages, posts, and groups. It can parse information including educational interests, access time preferences, language, media, and personal data. With massive amounts of rich data that are much larger than the data in a single isolated system and involve various types of user information, user profiling can achieve more accurate results than traditional datasets. Ostendorf et al. [39] combined social networks with conventional data-collection methods. They used volunteer data from Facebook and questionnaire survey data to obtain richer data sources. As a social networking site that focuses on photo sharing, Instagram has abundant photos. Therefore, several studies have used it as a data source. Stefanovič and Ramanauskaitė [17] used Instagram photos to make travel recommendations, whereas Dehshibi et al. [20] used photos on the platform to predict prototypical user requirements.

B. DATA PREPROCESSING

Most of the acquired raw data must be preprocessed before use. Preprocessing involves cleaning up non-compliant data, filtering duplicate data, and processing sparse samples. Some of these methods are summarized in Table 5.

Noisy data unrelated to the original data analysis affects the results and must be cleaned first. In this regard, Mobasher [37] noted that preprocessing should include removing irrelevant references from embedded objects, style files, graphics, or sound files, and invalid references due to spider navigation. The former task can be handled by analyzing server log references. The latter can be achieved by modifying the list of spiders and using heuristics or classification algorithms to reconstruct the Spiderman navigation model. Alhozaimi and Almishari [38] proposed three data preprocessing steps. The first step is filtering and cleaning, which are used to remove the noisy data. This step filters out retweets unrelated to the user, tweets containing images and videos, punctuation tweets, and @UserName tweets. The second step is normalization, which unifies the Arabic alphabet into a single form to overcome its diverse influences. The third step is tokenization, which breaks text into tokens.

Raw data are often messy and complicated to identify, and labeling is typically required. Tang et al. [40] used two preprocessing steps. The first step was to recognize the webpage marks. The webpage in this study had five marks: everyday words, unique words, pictures, terms, and punctuation. The second task was to assign tags to each mark according to the common word type, including location, contact, email, address, and phone. Photos and emails were assigned to the image tag, and the location was transferred to the term tag.

Preprocessing also involves processing unique characters. He et al. [41] proposed preprocessing methods for keywords such as template words and termination word removal. After preprocessing, keywords were extracted from the news text using LDA.

C. FEATURE EXTRACTION

Feature extraction is indispensable for user profiling because it requires a reasonable and sufficient input. The primary features include text, relationships, time, number, and image features. The common feature extraction methods are listed in Table 6.

Text features are the most popular. Alhozaimi and Almishari [38] extracted text styles and lexical features from Twitter content. Text-style features were selected based on user frequency. Lexical features were selected from Saudi

TABLE 5. Data preprocessing methods.

Data Preprocessing Approaches	References
Remove irrelevant references, style files, graphics, and sound files	[37]
Filtering, cleaning, normalization, and tokenization	[38]
Identify the marks on the web page and assign tags to each token	[40]
Delete template words to improve performance and remove all stop words from the input	[41]

TABLE 6. Feature extraction methods.

Feature Contents	Methods	References
Daily, weekday, and stay time access mode, token number and length, and number of each character type	Based on dataset observations, Google Web Search API, LDA	[12]
Text style and lexical features	Based on dataset observations	[38]
Final text features after weighted vector summation	CNN	[42]
Synthesis of three types of features: user contributions to topics, relationships between topics, and relationships between users	LDA, TagMe API	[43]
Visual features of the clothing	Database Generator, CNN	[44]

dialects. Moreover, machine learning approaches can extract deep features that are difficult to recognize. Zhu et al. [42] employed a convolutional neural network (CNN) to obtain features from the Weibo platform, where each tweet reflects a user's different life experiences and attitudes, and each message has a different degree of importance in determining user attributes. Researchers can also set additional weights for each post by introducing an attention mechanism. The weighted vectors are then summed for all texts to obtain the final user-text feature representation.

In user relation-based analyses, features are often represented as related information. Zarrinkalam et al. [43] comprehensively extracted three features: user contributions, relationships between topics, and user relationships to predict Twitter users' interests, and presented a heterogeneous graph and weighted undirected graph.

Time and identifier features were also used for user profiling. Fan et al. [12] designed the following feature extraction methods: First, four features were extracted for SSID-type analysis: daily access, weekday access, stay-time access, and access frequency modes. They can then map the SSID into five locations: workplaces, private places, dining places, shopping places, and entertainment places. Second, the following features were extracted for a given SSID: the number of tokens, average token length, delimiter number, and digit number. These features can determine whether the information encoded in SSID is informative.

When images are used for user profiling, the extracted features must be related to the photos, and the machine learning model must be combined to classify them. Reyes et al. [44] proposed three feature-extraction methods. The image processing method involves encoding the visual features of clothing pictures to extract the characteristic color of the dress. The images were divided into seven groups: pants, shoes, high heels, skirts, shirts, T-shirts, and sweaters. Color was extracted based on the image background. Label classification uses the VGGNET16 neural network to classify images between labels and obtain a confusion matrix. A formal-informal classifier was used to identify whether the clothes were formal. The photographs were divided into three groups: top, middle, and bottom. These three groups were classified as formal or informal. A total of 11 features were obtained using the above three feature extraction methods.

D. MODELING APPROACH

Many approaches are used in user modeling. Table 7 lists the models and their application scenarios, which are described in detail in the following section.

1) CLASSIFICATION AND CLUSTERING-BASED APPROACH

Classification and clustering are essential fields of machine learning, given their ability to perform excellently in specific tasks [45]. Several approaches have gained widespread use in user profiling and are highly effective in their respective roles.

a: K-MEANS CLUSTERING-BASED APPROACH

Data clustering analysis was used to divide data and ensure high similarity within clusters and low similarity between clusters [46]. The k-means clustering algorithm is one of the most representative data clustering algorithms [47].

The definition of the K-means algorithm is as follows: Given a dataset $X = \{x_i\}, i \in \{1, 2, \ldots n\}$, where i is a d-dimensional data point, X is divided into k clusters of $C = \{c_j\}, j \in \{1, 2, \ldots k\}$. The K-means algorithm aims to minimize the sum of the squared errors for each k cluster, and its formula is as follows:

$$J(c) = \sum_{i=1}^{k} \sum_{x_i \in c_k} \|x_i - \mu_k\|^2 \tag{1}$$

The operational steps of the k-means clustering algorithm [48] are shown in Algorithm 1.

The Euclidean distance formula is as follows:

$$\text{dist}_{ed}(x_i, x_j) = \sqrt{\sum_{u=1}^{n} |x_{iu} - x_{ju}|^2} \tag{2}$$

where x_i, x_j stand for the two points on the coordinates.

Some researchers have used K-means to conduct user-profiling research. Liao et al. [49] developed a framework for

TABLE 7. Data modeling technologies.

Modeling Type	Techniques	Description	References
Classification and clustering-based approach	K-MEANS	Developed a framework for data mining of Facebook users' behavior	[49]
		Improved K-means using canopy and mean calculation methods	[50]
		Proposed combining the collaborative filtering algorithm and K-means clustering to recommend movies to active users	[51]
		Combined K-means clustering and the Markov model algorithm to predict customer changes within a period	[52]
	K-NN	Used three different domain datasets: e-commerce, music, and news to test the user profiling and recommendation performance of K-NN	[55]
		Provided user identification based on different social network data	[58]
		Proposed a method to model user trustworthiness through trust and commitment in social networks	[59]
	Naïve Bayes (NB)	Conducted a study to perform sentiment analysis on Twitter users' reactions to the election debate	[61]
		Proposed a method to assess the credibility of jobseeker's information	[62]
		Proposed an aspect-based sentiment classification framework to extract useful information from travel reviews	[63]
	SVM	Created a personalized recommendation system to recommend attractions based on user preferences extracted from geotagged photos	[66]
		Proposed an SVM-based approach to identify depressed people	[67]
		Designed a model to identify fake profiles on Twitter	[68]
Artificial neural networks-based approach	MLP	Proposed a model for fake news detection in social networks	[71]
		Learned user preferences from behavioral data	[72]
		Proposed a location-aware personalized news model	[73]
	CNN	Considered the impact of similar friendships and provided interest recommendation	[6]
		Predicted user interest through text content in social networks	[75]
		Proposed a model to predict users' personality	[76]
		Predicted user personality from Twitter uploads and favorite photo content	[77]
		Proposed a model for predicting malicious accounts	[78]
	RNN	Proposed a context-learning model based on a hybrid gated recurrent neural network (GRNN)	[80]
		Explored varieties of strategies for integrating long-term user preferences and session pattern encoding	[81]
		Proposed a method MARC based on attribute embedding and recurrent neural networks	[82]
		Proposed a network attack detection framework that combines user behavior analysis and RNN	[83]
		Introduced a hybrid multi-feature framework for detecting fake news using RNN and GNN	[84]
Graph-based approach	GCN	Proposed a GCN with implicit associations model to obtain user profiles from social networks, which were represented as a heterogeneous graph	[86]
		Proposed a new graph convolution strategy to improve graph classification performance	[87]
		Designed a framework for transferring user profiles across social networks	[88]
	GAE	Proposed an ensemble clustering method based on cascading autoencoders for community detection	[91]
		Proposed an autoencoder-based model to extract overlapping communities in large networks	[92]
		Proposed a community detection method based on unsupervised deep learning	[93]
		Developed a deep adversarial substructure learning framework to learn	[94]

TABLE 7. *(Continued.)* **Data modeling technologies.**

	GAN	representations from user behavior graphs	
		Proposed a seedless graph deanonymization method	[95]
		Proposed an adversarial regularized graph embedding framework for community detection	[96]
Filtering-based approach	Rule-based filtering	Proposed that the information system set rules according to the information of registered users and filter users according to the rules	[100]
	Content-based filtering	Utilized users' existing profiles compared with new items for recommendation	[101]
		Introduced a content-based filtering system to recommend movies	[103]
		Proposed a CNN-based and content-based algorithm	[105]
	Collaborative filtering	Proposed a book recommendation system based on user and item collaborative filtering	[107]
		Proposed a collaborative filtering method based on movie reviews to recommend movies to users	[108]
		Proposed a position prediction framework based on users' historical trajectory	[109]
	Hybrid filtering	Proposed a hybrid filtering model to analyze users' opinions and infer their preferences	[114]
Knowledge-graph-based approach	Embedding-based	Proposed a learning method for embedding heterogeneous entities to analyze and recommend Amazon's e-commerce dataset	[117]
		Proposed a method to integrate the knowledge graph embedding algorithm into collaborative filtering for movie recommendation	[119]
	Connection-based	Proposed a method to calculate the correlation between users and items by extracting meta-paths in the knowledge graph	[121]
		Used meta-graphs to perform feature extraction on heterogeneous information networks	[122]
	Propagation-based	Proposed an approach to introduce a preference propagation mechanism in the knowledge graph	[116]
		Proposed a neighborhood interaction model to predict user preferences	[124]

Algorithm 1 K-means cluster

Input: Array $X\{x_1, x_2, \ldots x_n\}$
Output: Cluster number k, Cluster centroids $C\{c_1, c_2, \ldots c_k\}$
1: Initialize a set of k clusters:
2: $X = \{x_1, x_2, \ldots, x_n\}$
3: $C = \{c_1, c_2, \ldots, c_k\}$
4: **repeat**
5: **for** int $i = 1$ to n **do**
6: **for** int $j = 1$ to k **do**
7: Calculate the distance from a data point to the center of mass using the Euclidean distance formula as (2).
8: **end for**
9: Add data objects to the nearest cluster.
10: **end for**
11: Compute the new cluster centroid
12: **until** the difference between the cluster centroids obtained from two consecutive iterations remains unchanged.

the data mining of Facebook user behavior using K-means clustering based on a questionnaire survey of users' social media behavior and consumer preferences to divide users into three categories: K-pop boys, student blogs, and fan novels. Some researchers have also attempted to improve k-means clustering. Qian et al. [50] pointed out that the K-means algorithm has an unstable clustering number K value and initial centroid selection, leading to lower user classification accuracy and inaccurate label extraction. They then used canopy and mean calculation methods to improve the initial centroid selection and the K-means cluster number K. Other scholars have combined K-means with other techniques to create user profiles. Yassine et al. [51] proposed combining a collaborative filtering algorithm, with k-means clustering, to recommend movies to active users. This method first reduces the number of attributes through a dimensionality reduction method. K-means is then used to cluster movies into a specific number of classes. It then creates a profile file based on a set of parameters for each user. Finally, a collaborative filtering algorithm was used to select movies with the highest ratings for recommendation. Harish and Malathy [52] used historical retail transaction data combined with k-means clustering and Markov model algorithms to predict customer changes within a given period.

Although the K-means clustering–based technique has some limitations, such as selecting the value of cluster

number k, instability of the initial centroid, and user classification accuracy problem [50], [53], the simplicity and low computational complexity of the k-means clustering algorithm make it widely used in user profiling [48].

b: K-NEAREST NEIGHBOR–BASED APPROACH

The K-nearest neighbor (K-NN) is a supervised learning algorithm. With a given training dataset, a suitable subset of k users can be selected based on the similarity between k users and active users. Subsequently, the prediction of active users can be generated according to the weighted aggregation of their scores [54]. According to the different goals of the similarity comparison, K-NN can be divided into two categories, IKNN and SKNN, which are explained below.

IKNN considers the last behavior within a given session [55]. These are typically used in real-time scenarios. The model returns k items that are most similar to the behavior as recommendations, and other items as follow-up choices. Each item was represented as a binary vector containing a code of 0 or 1. Each item in the binary vector corresponded to a specific session image. If an item exists in a session, the value is set to one; otherwise, it is set to zero. The similarity between two items was computed from binary data using cosine similarity measurement technique. k is the number of nearest neighbors, and the value of k is defined according to the length required for the recommendation. The similarity calculation between items I_a and I_b is formulated as (3):

$$\mathrm{Sim}\,(I_a, I_b) = \mathrm{Cos}\,(I_a, I_b) = \frac{I_a \cdot I_b}{||I_a|| \cdot \cdot ||I_b||}. \tag{3}$$

SKNN takes the entire current session and compares it to all previous sessions to find the k most similar sessions [56]. Compared with IKNN, it uses the entire session; therefore, it obtains more information and more accurate recommendation results, and is suitable when more accurate recommendation results are required. Session similarity was measured using cosine similarity and the Jaccard index. The cosine similarity was calculated using equation (4), and the Jaccard similarity coefficient was calculated using equation (5). Finally, the similarity result was calculated to measure the recommendation score of the candidate item relative to the current session using (6).

$$\mathrm{Sim}\,(c, S_i) = \mathrm{Cos}\,(\mathrm{c}, S_i) = \frac{c \cdot S_i}{\|c\| \cdot \|S_i\|} \tag{4}$$

$$J(c, S_a) = \frac{c \cap S_i}{c \cup S_i} \tag{5}$$

$$\mathrm{Score}_{sknn}(v, c) = \sum_{S_{nb} \in N_c} \mathrm{Sim}(c, S_{nb}) \cdot l_{S_{nb}}(v) \tag{6}$$

where c represents the current session, and N_c represents the set of nearest-neighbor sessions in the current session. S_{nb} represents any session in N_c except c. $Sim(c, S_{nb})$ represents the candidate recommendation item similarity between c and S_{nb}. v is the candidate recommendation item. $l_{S_{nb}}$ represents a function set to a value of one when item v exists in S_{nb}; otherwise, it returns zero.

The SKNN method does not consider the order of elements within the session. Some extensions of the SKNN method have been proposed, such as S-SKNN and SF-SKNN, which are most suitable when the order of elements may be relevant in some domains, and user preferences may be related to the order in a session.

In the S-SKNN algorithm [57], items that appear later in a session are assigned more weights. In these cases, (6) requires the addition of more parameters to obtain the following:

$$\mathrm{Score}\,(v, c) = \sum_{S_{nb} \in N_c} \mathrm{Sim}\,(c, S_{nb}) \cdot \omega_{S_{nb}}\,(c) \cdot l_n\,(v) \tag{7}$$

where some parameters are the same as those in (6) and $l_n(v)$ stands for an indicator function affected by $\omega_{S_{nb}}(c)$. By contrast, $\omega_{S_{nb}}(c)$ represents a weighting function that is positively correlated with the degree of recentness of the session. We assumed that C_p is the most recent item in the current study. Then, $\omega_{S_{nb}}(c)$ can be described as:

$$\omega_{S_{nb}}\,(c) = \frac{p}{|c|} \tag{8}$$

where p stands for the position of C_p.

SF-SKNN uses more constraint score functions, meaning that it only recommends one item that appears exactly after the last interaction in the current session. It is calculated as

$$\mathrm{Score}_{sf-sknn}\,(v, c) = \sum_{S_{nb} \in N_c} \mathrm{Sim}\,(c, S_{nb}) \cdot l_n\left(c_{|l|}, v\right). \tag{9}$$

Most of the score functions were the same as those of the SKNN. The indicator function $l_n(c_{|l|}, v)$ differs. Suppose that $c_{|l|}$ appears in the user's current session c; only when c contains a sequence $(c_{|l|}, v)$, l_n return 1.

Several researchers have explored this topic. Zhou et al. [58] used the k-NN technique to create a user identification scheme that takes advantage of the reliability and consistency of friendships in different social networks. This solution initially models the network structure and uses network-embedding technology to embed each user's characteristics into a vector, which converts the user-identification problem into a nearest-neighbor problem. Finally, a scalable k-NN algorithm is proposed to calculate and match users. Berkani et al. [59] used Yelp to develop an approach to model users' credibility based on their trust in and commitment to social networks. Two separate classification techniques were used. The K-means algorithm was applied to all users, and the k-NN algorithm was used for newly added users.

The k-NN-based technique is simple to implement and achieves a good performance [55]. Nevertheless, k-NN cannot achieve good results for unbalanced data and is easily affected by noisy data [54].

c: NAÏVE BAYES-BASED APPROACH

The Naïve Bayes (NB) algorithm is based on the popular Bayes theorem, and is a commonly used probabilistic classification technique in MLDA (machine learning and data analytics). It is simple, effective, and robust [60]. The following are some examples of researchers using NB.

Pratama et al. [61] conducted sentiment analysis of Twitter users' reactions to election debates in Jakarta, Indonesia. The NB algorithm was used to obtain the sentiment values of residents within and outside Jakarta for the three candidates.

In this study, the user tweets were preprocessed, trained, and classified into unknown category documents. The general formula for Bayes' theorem is as follows:

$$P(H|X) = \frac{P(X|H)\,XP(H)}{P(X)} \tag{10}$$

where $P(H|X)$ represents the posterior probability of assuming that H occurs when given evidence E. $P(X|H)$ represents the probability that the occurrence of evidence E will affect hypothesis H. $P(X)$ represents the prior probability evidence for E.

Because it is difficult to determine which word in the sentence is a feature word, the author assumed that each word is a feature and then applied Bayesian theory. The formula used was as follows:

$$P(K|F) = \frac{P(F|K)\,XP(K)}{P(F)} \tag{11}$$

where F represents a feature, and K represents a category.

Many features or words may support the same category, such as features F_1, F_2, and F_3. Thus, (11) can be expanded as follows:

$$P(K|F_1, F_2, F_3) = \frac{P(F_1, F_2, F_3|K)\,XP(K)}{P(FF_1, F_2, F_3)} \tag{12}$$

Because Bayesian theory requires that the existing evidence be independent of each other, (12) can be changed as follows:

$$P(K|F_1, F_2, F_3) = \frac{P(F_1|K)\,XP(F_2|K)\,XP(F_3|K)\,XP(K)}{P(F_1)\,XP(F_2)\,XP(F_3)} \tag{13}$$

If a general description is used, (13) can be expressed as follows:

$$P(K|F) = \frac{\prod_{i=0}^{q} P(F_1|K)}{P(F)}. \tag{14}$$

Jing et al. [62] proposed a method to assess the credibility of information about candidates recruited on the internet. The authors used a tree-augmented NB (TAN) classifier to calculate the credibility probability of user-analysis information. Based on the PageRank algorithm, they measured the authority of individual users by analyzing user interactions in professional social networks. Finally, the proposed method was validated using LinkedIn user profiles. Afzaal et al. [63] proposed an NB-based sentiment classification framework that extracts useful information from travel reviews and performs restaurant or hotel classification tasks to help tourists find good restaurants or hotels in the city.

NB performs well when the assumptions are met, and for smaller-sized datasets, it often outperforms alternative techniques [64].

d: SUPPORT VECTOR MACHINE–BASED APPROACH

Support vector machines (SVMs) follow the structural risk minimization principle to construct an optimal hyperplane that separates data points of positive and negative data examples with the broadest possible interval [65]. SVM has been widely and successfully used for user profiling.

Sun et al. [66] downloaded a geotagged photo collection from Flickr.com and created a personalized recommendation system to recommend attractions based on user preferences. The generation of candidate attractions is a multiclassification problem. The authors used the SVM algorithm to generate an attraction candidate list $L_{candidate}$ for the corresponding user, according to the user's travel history attraction collection L_u. Candidates include attractions that users are most likely to visit. Formally, the process of SVM segmenting data points is expressed as $\omega^T x + b = 0$, where ω represents a vector perpendicular to the hyperplane, and b represents the offset of the hyperplane. The training process of SVM can be understood as a hyperplane optimization problem. The objective function of the problem is expressed as follows:

$$\max \frac{1}{||\omega||}, \quad s.t.,\ y_i\left(\omega^T x_i + b\right) \geq 1, \quad i = 1, \ldots, n \tag{15}$$

To solve the optimization problem with constraints, researchers applied the Lagrange multiplier method to the objective function. The Lagrangian dual equation of Equation (15) is as follows:

$$\max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j,=1}^{n} \alpha_i \alpha_j y_i y_j x_i, x_j$$

$$s.t.,\ 0 \leq \alpha_i \leq C, \quad i = 1, \ldots, n, \sum_{i=1}^{n} \alpha_i y_i = 0 \tag{16}$$

where α_i denotes the Lagrange multiplier for each data point. C represents the slack variable that penalizes misclassified data, and $\langle x_i, x_j \rangle$ are the inner products of vectors x_i and x_j. Because SVM is a binary value classifier, to support multiple outputs in recommendation application scenarios, the authors used the one-vs-rest (OvR) strategy to convert a multiclassification problem into a binary classification one.

The training process of SVM is shown in Algorithm 2.

Here, U_{train} represents the collection of all users, $DataSet_{SVM,train}$ is a collection of (attraction tags) that users have visited historically, and L represents the collection of attraction tags.

After inputting the test user's historical attraction $L'_{u'}$, the attraction confidence value $Prob_l$ can be obtained by predicting the candidate's attractions. Candidate $L_{candidate}$ is then generated from the top n values of $Prob_l$.

However, other studies have obtained different results. Peng et al. [67] proposed an SVM-based approach for identifying individuals with depression. The author first extracted three types of features: user Weibo text, user profile, and user behavior from the user's social media, and then used a multi-kernel SVM method to adaptively select optimal kernels for different features to discover depressed users.

Algorithm 2 The Training Process of SVM

Input: U_{train}, L, $DataSet_{SVM,train}$
Output: *Multiclass_SVM_Classifier*
1: **for each** l **in** L **do**
2: **for each** $(L'_{u'}, label)$ **in** $DataSet_{SVM,train}$ **do**
3: **Encode** $L'_{u'}$ **into** *Vec*
4: **if** $(label == 1)$
5: **Add** *(Vec, 1)* **to** $DataSet_l$
6: **else**
7: **Add** *(Vec, 0)* **to** $DataSet_l$
8: **end if**
9: **end foreach**
10: $SVM\_Classifier_l$ **fit** $DataSet_l$
11: **Add** $SVM\_Classifier_l$ **to** *Multiclass_SVM_Classifier*
12: **end foreach**

Kadam and Sharma [68] designed a model for identifying fake Twitter profiles. The model first improves the quality of the dataset by refining the data content and attributes. They also used a model to reduce computational complexity by reducing data dimensionality. The model predicts fake users using SVM and other classification algorithms.

SVM has been widely used as powerful tools in various user-profiling applications. It is suitable for processing classification problems of high feature dimensions, dense concepts, and sparse instance data, especially for text data [65].

2) ARTIFICIAL NEURAL NETWORKS–BASED APPROACH

Artificial neural networks are becoming increasingly crucial for user profiling [21]. These networks offer many advantages for processing large volumes of data. However, they require access to numerous labeled datasets and powerful computing and storage capabilities to be genuinely effective [69].

a: MLP-BASED APPROACH

An MLP is a relatively simple artificial neural network comprising multiple fully connected layers of neurons. In addition to the input and output layers, each layer is simultaneously connected to the previous and subsequent layers, receiving input from the previous layer, and passing the output to the next layer [70].

MLP is now primarily used as part of an entire model and cooperates with other algorithms to solve problems. Ali et al. [71] proposed a model for fake-news detection using social networks. The model utilizes the MLP for decision making. The MLP in this model involves three layers: the input layer, hidden layer, and output layer. The ReLU function was used to complete the activation of neurons, and the softmax function was used for classification. The stacked layers of the MLP used input features. The fake news classifier can be described as follows: ω_i represents the weight of neuron item i, θ represents the weights of the deep learner trained in the previous stage, and x_i represents the output of the previous layer. The score of correct prediction $p(c)$ can then be formulated as follows:

$$p(\text{class\_label} = c) = \sum_{i=1}^{n} \omega_{ci} * x_i + \theta. \tag{17}$$

Then, the logistic function is computed as follows:

$$\text{logit}\left(p\left(\text{class}_{label} = c\right)\right) = \log\left(\frac{p(\text{class}_{label} = c)}{1 - p(\text{class}_{label} = c)}\right) \tag{18}$$

where the x_c logit represents the logistic predicted class function, which maps values from the range $(-\infty, +\infty)$ into [0, 1]. Suppose that $\vec{x} = \{x_1, x_2, x_3, \ldots, x_c\}$ vectors contain the predicted scores (x_c) of class c. The final predicted class label can be calculated according to the softmax function, as follows:

$$\forall x_c \in \vec{x} \text{ calculate } \frac{e^{x_c}}{\sum_{c=0}^{k} e^{x_c}}) \rightarrow \vec{V} \tag{19}$$

where $\vec{V}$ denotes a vector that determines the predicted class label by determining the maximum probability. The predicted class is calculated as follows:

$$p = \text{index\_of\_max}(\vec{V}) \tag{20}$$

where p is the predicted class.

Other scholars have also conducted research in this field. Gao et al. [72] proposed a new multitask recommendation approach using MLP to predict user preferences based on various behavioral data (e.g., browsing and clicking). This approach relates the model predictions for each behavior type in a cascaded manner to capture the sequential relationships between the behavior types. Chen et al. [73]proposed a location-aware personalized news recommendation system based on MLP and deep semantic analysis. The system uses deep neural networks to map the topic space based on the Wikipedia concept to an abstract, dense, and low-dimensional feature space that maximizes the similarity between users and target news.

MLP, which is suitable for processing low-dimensional data such as tabular data, has difficulty supporting high-dimensional data such as photo datasets because this scenario makes the number of fully connected layer parameters too large, thus making model training difficult [69].

b: CNN-BASED APPROACH

A CNN is a class of feedforward neural networks with profound structure and convolutional computations [74]. It is suitable for high-dimensional deep-learning tasks, such as image recognition.

CNN can also be used to charge text. Kang et al. [75] proposed an approach for predicting user interest in social networks. In this framework, several phases exist before the CNN. First, the words in the social network are mapped into vectors used as input to a bidirectional gated recurrent unit (biGRU) by the word-embedding technique. Second, a sentencing matrix is constructed using the output of the

biGRU as the input to the CNN model. The CNN can then predict user interests. The CNN model includes convolutional layers, activation units, max-pooling layers, and a softmax layer.

In the convolutional layer, k filters are applied to each message of length h. After random initialization, each filter performs a role according to its weight during the training. In classification operations, filters can detect phrases or sentences that are related to a topic. The feature generation operation is formulated as follows:

$$c_i = \sum_{l,m} (X_{[i:i+h]})_{l,m} \cdot F_{l,m} \tag{21}$$

where $c_i \in R^{n-h+1}$ represents a feature vector, $F \epsilon R^{h \times d}$ represents a filter, and X_i represents a word vector. A feature map matrix $c \in R^{k \times (n-h+1)}$ consists of all c generated from all filters k.

A nonlinear activation function follows each convolutional layer, thereby enabling nonlinear decision-boundary learning. To improve the training speed and accuracy, researchers used ReLU as the activation function in the model.

The information is aggregated, and the representation is reduced in the pooling layer to fix the output matrix. In other words, the pool layer minimizes the dimensionality of the output while preserving valuable features. The operation in the pooling layer is formulated as:

$$c_{pooled} = \begin{pmatrix} \text{pool}(\alpha(c_1 + b_1 * e)) \\ \cdots \\ \text{pool}(\alpha(c_n + b_n * e)) \end{pmatrix} \tag{22}$$

where c_i is the ith convolutional feature map c, b_i is a bias term, e is a unit vector of the same size as c_i, and function α is the activation function. The max-pool choice, which only returns the maximum pool value, is adopted in this model.

The fully connected softmax layer is the final layer that uses the softmax function to classify the multiclass data passed by the pooling layer. The probability distribution in Softmax is formulated as follows:

$$\begin{aligned} P(y = j \mid x, b) &= \text{softmax}_j \left(x^t \omega + b\right) \\ &= \frac{e^{x^T \omega_j + b_j}}{\sum_{k=1}^{J} e^{x^T \omega_k + b_k}} \end{aligned} \tag{23}$$

where ω_j and b_j are the weight vector and the bias of the jth class, respectively.

Convolutional neural networks (CNNs) are commonly used for image recognition. Cucurull et al. [76] proposed a model to predict a user's personality according to the Big Five personality traits using pictures on social networks. The model's input was images to be classified from social networking sites, and the output was one of five personalities. The model consists of multiple layers, where the last layer is a classifier that projects the input features into class labels. The model can be represented as:

$$f(x; \theta) = f^n(f^{n-1}(\ldots f^2(f^1 x; \theta_1); \theta_2); \theta_{n-1}); \theta_n) \tag{24}$$

where x is an input image, the nonlinear function $f(x; \theta)$ maps the image features to the output, CNN has multiple layers, and N represents the number of layers in CNN. θ_n represents the parameters of each layer of the nonlinear function, which is formulated as:

$$\theta = \text{argmin}_\theta L\left(y, \hat{y}\right) \tag{25}$$

where L is the loss function and y and $\hat{y}$ represent the predicted and actual outputs, respectively. The loss function is calculated iteratively using the stochastic gradient descent method.

Because the model predicts five personality types and each personality result has two results (yes and no), each personality probability can be formulated as:

$$p_i(o_i) = \frac{e^{o_i}}{\sum_{j=1}^{2} e^{o_i}} \tag{26}$$

where vector o is the softmax function input and p represents the score vector of a particular personality possibility.

Considering five personalities, to reduce the number of calculations, the model expresses the loss function L as

$$L_c = \begin{cases} -\log(p_l) & \text{correct classifier} \\ 0 & \text{ignored classifier} \end{cases} \tag{27}$$

where the classifier is zero if no classifier needs to be considered.

Other scholars have also conducted research in this field. Safavi and Jalali [6] proposed a new point-of-interest recommendation method based on deep learning and CNN. This approach only considers the impact of the most similar friendships. The spatial and temporal features of the most similar friends are used as the input of the CNN model, and the output of the CNN can be used to predict the latitude and longitude and the subsequent appropriate location ID (identification). Finally, the friendship interval of a similar pattern is used to select the location using the smallest distance method. Guntuku et al. [77] used a 19-layer VGG network image classifier based on a CNN to predict the top five personalities of Twitter users using the ImageNet tag set of Twitter users. Wanda [78] proposed a model to predict malicious accounts. The model trained the data by dynamically building a CNN and using activation layers instead of traditional functions to improve the accuracy of the training and testing processes.

CNNs can provide better accuracy and enhance the system performance through local connectivity and shared weights. Furthermore, they perform well in natural language processing and computer vision but often perform poorly when the input data depend on each other in a sequential pattern [79].

c: RNN–BASED APPROACH

An RNN can process sequence information better than a CNN can. It stores past information and current inputs by introducing state variables to predict current output [69].

The recurrent neural network (RNN) algorithm is widely used for user profiling. Cui et al. [80] proposed a rich

context-learning model based on a hybrid-gated recurrent neural network (GRNN). This model can be used as an example to explain RNN. This model combines static, post-time, sequential, part-of-speech (POS) tags, and historical features to achieve better prediction results. The authors propose a hybrid model based on previous joint, hierarchical, and contextual models. LSTM and GRU are both types of RNNs. Both consider the order and dependencies of the tokens. The authors described both models in this study. The formula for the hybrid GRU (HD-GRU) can be represented as

$$\begin{aligned} z_t &= \sigma\left(W_{xz}x_t + W_{hz}h_{t-1} + b_z + W_{Ez}E\right) \\ &= \sigma\left(\left[W_{xz}W_{Ez}W_{hz}l\right]\left[x_tEh_{t-1}b_z\right]\right) \end{aligned} \tag{28}$$

$$\begin{aligned} r_t &= \sigma\left(W_{xr}x_t + W_{hr}h_{t-1} + b_r + W_{Er}E\right) \\ &= \sigma\left(\left[W_{xr}W_{Ez}W_{hz}l\right]\left[x_tEh_{t-1}b_r\right]\right) \end{aligned} \tag{29}$$

$$\begin{aligned} \tilde{h}_t &= \tanh\left(W_{xh}x_t + W_h\left(r_t * h_{t-1}\right) + b_h + W_{Eh}E\right) \\ &= \tanh\left(\left[W_{xh}W_{Eh}W_hl\right]\left[x_tE\left(r_t * h_{t-1}\right)b_h\right]\right) \end{aligned} \tag{30}$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \tag{31}$$

where z is the update gate; r is the reset gate; x is the input; b is the bias; $\tilde{h}$ and h represent the candidate value for the hidden state and hidden state, respectively; W represents the corresponding weights; and E refers to the contextual feature. The structure of the hybrid GRU is illustrated in Fig. 4.

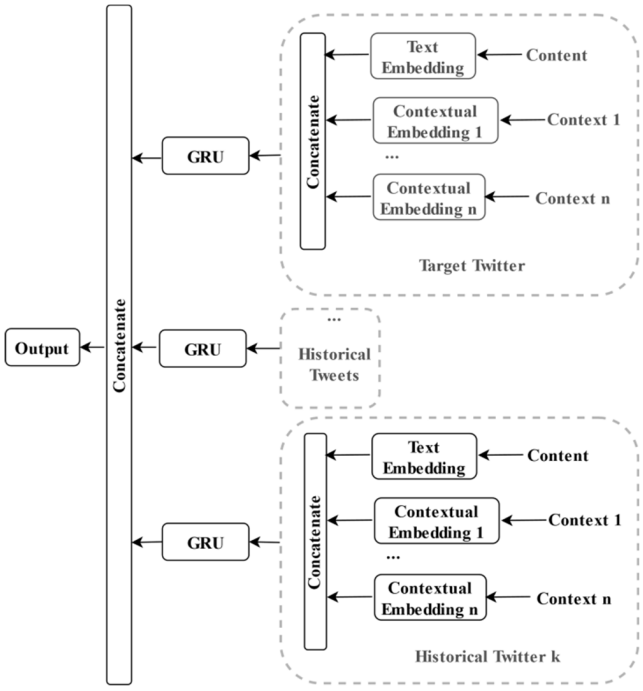

FIGURE 4. Structure of hybrid GRU [80].

Other scholars have also conducted research in this field. Phuong et al. [81] proposed using multiple strategies to integrate long-term user preferences and session patterns with RNN, including the incorporation of user embeddings and contributions using specially designed gating mechanisms. After an empirical evaluation of the test dataset, it was found that combining the long-term user profile with the output of the session RNN improved prediction results. May Petry et al. [82] proposed a method called MARC based on attribute embedding and RNN to address the complexity of heterogeneous data dimensions and the timing of motion features. This method can classify heterogeneous trajectories and handle all trajectory attributes including space, time, semantics, and sequences. This method performs well for various text and category attribute descriptions. In the field of RNN applications, Alshehri et al. [83] proposed a network attack detection framework that combined user behavior analysis and machine learning. The framework involves mapping user behavior onto a sequence of network events. Subsequently, a recurrent neural network is employed to identify and categorize these behaviors as regular or irregular based on their extracted features. Chalehchaleh et al. [84] introduced a hybrid multifeature framework for detecting fake news. The framework comprehensively considers news text, user profiles, and pictures. It uses a combination of pretrained language models, RNN, and GNN, to analyze and make predictions about fake news.

RNN methods have feedback loops in their recurrent layers, which help them retain information in ''memory'' for a long time, so they perform superbly in sequence-learning tasks. However, training a standard RNN to solve this problem is time-consuming. Dependent learning problems can be challenging because the loss function gradient decays exponentially over time; this is also known as the vanishing gradient problem [79].

3) GRAPH–BASED APPROACH

The graph-based user profiling approach is a commonly used method that uses graph heterogeneity, diversity, and interdisciplinary characteristics. This approach can represent objects and relationships in various fields such as social networks, transportation networks, and biological networks. Analyzing graphs can provide detailed user profiling, which is also widely used.

a: GRAPH CONVOLUTIONAL NETWORK-BASED APPROACH

A graph convolutional network (GCN) [85] is a semi-supervised learning method that extracts the features of known nodes and graph structures using a CNN to infer the classification of unknown nodes. GCN are widely used in research fields such as node classification, graph classification, and edge prediction. Wen et al. [86] proposed a GCN with implicit associations (GCN-IA) model to obtain user information from social networks, which were represented as heterogeneous graphs. The model comprises the following three modules:

1. The prior knowledge enhancement (PKE) module captures the implicit association between tags to obtain the connection between users and profiles for user representation.

2. The user representation module jointly learns user and tag embeddings based on the text, relations, and labels.
3. The classification module performs classification and predicts user profiles.

In the PKE module, the prior knowledge probability matrix P is formulated as follows:

$$P_{ij} = \frac{\left|\left\{t|t \in I \wedge \left(l_i, l_j\right) \leq t\right\}\right|}{\sum_{i=0}^{m}\sum_{j=0}^{m}\left|\left\{t|t \in I \wedge \left(l_i, l_j\right) \leq t\right\}\right|} \tag{32}$$

where P_{ij} is higher, and the propagation probability between labels is greater.

Considering the complexity of social relationships, researchers have defined tag sets in the model as follows:

$$I = I_1 \cup I_2 \cup I_3 \cup \cdots \tag{33}$$

where $I_i (i = 1, 2, 3, \ldots)$ represents the interest tag set of each user.

A GCN is a multilayer neural network that is used to learn iterative convolution operations in graphs. The model applies a GCN to embed user information and personal tags into the vector space, and learns user and tag representations from user-generated content and social relationships. Formally, the model uses a social network $G = (V, E)$, where V and E represent sets of nodes and edges, respectively. After the model is initialized, edges are constructed between nodes based on the implicit associations among user relationships, user profiles, and tags. The model introduces adjacency matrix A and its degree matrix D, where $D_{ij} = \sum_{j=1,\ldots,n} A_{ij}$. The weight of the edge between the user and label nodes is expressed as:

$$A_{ij} = \begin{cases} 1 & \text{if the user i is with the label } j, i \in U_{gold}, j \in C \\ 0 & \text{otherwise} \end{cases} \tag{34}$$

where U represents the user set in the social network, and U_{gold} represents the labeled user. C is the label set of user profiles.

To enhance knowledge, the model must calculate the weight between two tag nodes as follows:

$$A_{ij} = \begin{cases} 1 \times \text{sim}\,(i, j) & \text{if } \left(u_i, u_j\right) \in R \\ 0 & \text{otherwise} \end{cases} \tag{35}$$

where $R = \left\{\left(u_0, u_1\right), \left(u_1, u_3\right), \ldots\right\}$ is the user relationship set, and $sim(i, j)$ represents the similarity between users i and j. For a single-layer GCN, the new node feature matrix can be expressed as

$$L^{(1)} = \sigma\left(\tilde{A}XW_0\right) \tag{36}$$

where $\tilde{A}\left(\tilde{A} = D^{-1/2}WD^{-1/2}\right)$ is the normalized symmetric adjacency matrix, W_0 is the weight matrix, and $\sigma(\cdot)$ is the activation function. By stacking GCN layers, the model calculates the information propagation process as follows:

$$L^{(j+1)} = \sigma\left(\tilde{A}L^{(j)}W_j\right) \tag{37}$$

where j represents the number of layers, $L^{(0)} = X$.

The authors framed user interest prediction as a multiclassification problem. Based on the previous steps, the model obtains the user representation based on user-generated content and relationships, embeds user-represented nodes into the softmax classifier, and projects the final representation as follows:

$$Z = p_i\,(c|R, U; \Theta) = \text{softmax}\left(\tilde{A}\sigma\left(\tilde{A}XW_0\right)W_1\right). \tag{38}$$

The authors used user cross-entropy error as a loss function, expressed as follows:

$$L = -\sum\nolimits_{u \in y_u}\sum\nolimits_{f=1}^{F} Y_{df} \ln Z_{df} \tag{39}$$

where y_u is the set of labeled user indexes and F is the dimensionality of the output features. Y is the label indication matrix.

Other scholars have also conducted research in this field. Pasa et al. [87] proposed a new graph convolution strategy to improve the graph classification performance, which considers a single graph convolution layer that independently utilizes neighboring nodes at different topological distances to generate decoupled representations. The subsequent readout layers process these representations. To implement this strategy, the authors introduced polynomial graph convolutional layers (PGC). Diao et al. [88] focused on the problem of transfer learning between social networks and designed a framework for transferring user profiles across social networks that can transfer user relationship knowledge between data-rich and data-scarce social networks. The authors first designed a GCN based on the feature-aware domain attention model to discover user dependencies inside and outside the social network. They designed an adversarial learning model to solve the domain drift problem during the migration process.

The graph convolution layer of the GCN model is Laplacian smoothing, which mixes the characteristics of nodes and their domains, giving the phase neighboring nodes similar features and greatly simplifying the classification task. Nonetheless, the output features are too smooth; therefore, the nodes in different clusters cannot be effectively distinguished. In addition, the GCN model is difficult to expand, and thus, cannot efficiently handle large-scale directed graphs [89].

b: GRAPH AUTOENCODER–BASED APPROACH

Graph autoencoders (GAEs) can effectively represent nonlinear networks, and deep neural network–based autoencoders can learn representations of datasets in an unsupervised manner [90]. Xu et al. [91] proposed an ensemble clustering method based on cascading autoencoders (CDMEC) for community detection, which can aggregate nodes in a social network into a series of substructures. First, the authors constructed four similarity matrices to describe the local information of nodes comprehensively. Second, the authors used transfer learning methods to discover shared parameters.

An autoencoder consists of an encoder and a decoder. The encoder converts the input data into a hidden layer feature as follows:

$$\xi^{(r)} = S\left(Wm_i^{(r)} + b\right) \tag{40}$$

where $m \in R^N$ represents the input data, $\xi \in R^d$ represents the hidden layer feature, $S(x)$ represents a nonlinear activation function, W is the weight matrix, $r \in \{s, t\}$ represents the data dataset, and b represents bias.

The decoder converts the hidden layer features $\xi^{(r)}$ to the reconstruction $\hat{m}^{(r)}$ as follows:

$$\hat{m}^{(r)} = S\left(\hat{W}\xi^{(r)} + \hat{b}\right) \tag{41}$$

where $\hat{W}$ is the weight matrix, and $\hat{b}$ is the bias.

Under parameter $\theta = \{W, \hat{W}, b, \hat{b}\}$, by minimizing the reconstruction error, the weight matrices W and $\hat{W}$ can be learned through an iterative parameter-updating algorithm. The objective function is expressed as follows:

$$J^{(r)}(\theta) = \min_{\theta} \sum_{i=1}^{N^{(r)}} L\left(m_i, \hat{m}_i\right) + \alpha \sum_{j=1}^{s^{(r)}} \rho \log \frac{\rho}{\hat{\rho}_j} + (1-\rho) \log \frac{1-\rho}{1-\hat{\rho}_j} \tag{42}$$

where α represents the weight coefficient, $L\left(m_i, \hat{m}_i\right)$ represents the distance function, s represents the neuron number, j represents the number of hidden layer neurons, and ρ represents a sparsity parameter that is close to zero. The parameter training process was as follows:

$$\begin{aligned} W &\leftarrow W - \gamma \frac{\alpha J(\theta)}{\alpha W}, \quad b \leftarrow b - \gamma \frac{\alpha J(\theta)}{\alpha b} \\ \hat{W} &\leftarrow \hat{W} - \gamma \frac{\alpha J(\theta)}{\alpha \hat{W}}, \quad b \leftarrow \hat{b} - \gamma \frac{\alpha J(\theta)}{\alpha \hat{b}} \end{aligned} \tag{43}$$

where γ represents the learning rate.

Finally, there is an ensemble clustering framework that consists of generation and identification. Considering its simplicity and speed in generation, the author used the k-means clustering algorithm to obtain the essential clustering result consistency matrix Q, which can be used to detect clustering results using the NMF-based clustering method during identification. This process can be formulated as follows:

$$\min_{H \geq 0} D_E(Q \| WH) = \min_{H \geq 0} \frac{1}{2} \|Q - WH\|_2^2 \tag{44}$$

where $D_E(Q|WH)$ represents a cost function based on the Euclidean distance, W represents a nonnegative basis matrix, and H represents a coefficient matrix.

During the optimization process, the multiplicative update rule is used to estimate H to ensure non-negativity of the optimization results. The update rules for w_{ik} and h_{ik} can be obtained as follows:

$$\begin{aligned} w_{ik} &\leftarrow w_{ik} \frac{(QH^T)_{ik}}{(WHH^T)_{ik}} \\ h_{ik} &\leftarrow w_{ik} \frac{(QH^T)_{ik}}{(WHH^T)_{ik}} \end{aligned} \tag{45}$$

where $h_{i,j}$ stands for the correlation between the ith sample and the jth category.

Then, the final clustering result matrix is obtained as follows:

$$E_K = \left\{ \{C_1, \ldots, C_k\} : v_i \in C_j, if\ h_{i,j}^* = 1 \right\} \tag{46}$$

where $i = 1, \ldots, n$ and $j = 1, \ldots, c$.

Other researchers have conducted studies in the field of graph autoencoders. Bhatia and Rani [92] proposed a model that resolves overlapping communities in large networks using autoencoders. The model consists of three stages. In the first stage, the seed was selected using an autoencoder. The second stage aims to learn the community topology through seed expansion. The third stage obtains a better representation of the refined clusters. Hao and Zhang [93] proposed an unsupervised user-user graph detection method. First, the edge weights of the user graph are calculated based on the similarity of user behaviors. Second, a cascaded denoising autoencoder extracts features, generates clustering results, and reconstructs the graph. Third, a persistence optimization algorithm is used for community detection.

The GAE-based technique can embed nonlinear characteristics and map data points to a low-dimensional space for dimensionality reduction. It is suitable for unsupervised learning fields such as community detection [90].

c: GENERATIVE ADVERSARIAL NETWORK-BASED APPROACH

A generative adversarial network (GAN) employs two competing deep neural networks: a generator and discriminator. The former generates samples, whereas the latter ensures that the samples originate from prior data distribution [98].

Some researchers have used a graph-based GANs for user profiling. Wang et al. [94] studied the problem of using the point of interest check-in data to evaluate mobile users. The authors first constructed a graph representing each user and then proposed a deep adversarial substructure learning framework to learn representations from user-behavior graphs.

The framework comprises an autoencoder, approximate substructure detector, discriminator, and adversarial trainer. Autoencoders preserve the overall structure of a graph and derive their representations. A subgraph detection algorithm was used to detect the subgraph labels, and these label pairs were pretrained using a CNN. A discriminator was used to classify the substructures of the original and reconstructed graphs. The adversarial trainer forces the autoencoder to preserve the substructure in the reconstructed graph, thus confusing the discriminator and incorporating substructure awareness. Subsequently, optimization problems must be solved, and user representations are finally obtained for activity-type prediction.

The encoding converts the input user activity graph into user feature vectors, which can be decoded to reconstruct the graph and capture the global behavioral structure by minimizing the reconstruction loss between the original image x and the reconstructed image $\hat{x}$. This process can be expressed as

follows:

$$L_{AE} = \frac{1}{2}\sum_{i=1}^{m} |\left|(x_i - \hat{x}_i)\right|\,|_2^2. \tag{47}$$

The framework uses pretrained CNNs to emulate traditional substructure detectors. First, it generates substructures (labels) and then trains the CNN-based detector F_{cnn} to be close to the traditional substructure F_{detr}, making it a differentiable approximate substructure detector. Let $\hat{s}$ represent the output of F_{cnn}, which can be expressed as

$$L_{cnn} = \frac{1}{2}\sum_{i=1}^{m} \left\|(s_{real} - \hat{s})\right\|_2^2. \tag{48}$$

The adversarial learning strategy in the framework involves a generator, discriminator, and adversarial trainer. The generator connects a deep autoencoder with a pretrained CNN detector. The reconstructed image $\hat{x}_i$ was used as the input to the CNN. The CNN then generates and outputs the substructures represented by $\hat{s}_i$. Assuming G represents a generator, the mapping process is as follows:

$$\hat{s}_i = G\left(\hat{x}_i\right). \tag{49}$$

The discriminator is an MLP that outputs a probability, indicating the likelihood that a substructure is from a real, and not a generated, set of substructures.

The framework simultaneously trains the parameters G and D in the adversarial training strategy. D is used to maximize the classification accuracy of the real substructures, and G is used to minimize D's classification of the reconstructed substructures generated by G. The accuracy L_D of the discriminator was formulated as follows:

$$L_D = \frac{1}{m}\sum_{i=1}^{m} [\log D\left(s_i\right) + \log\left(1 - D\left(G(x_i)\right)\right)]. \tag{50}$$

The generator loss L_G can be expressed as follows:

$$L_G = \frac{1}{m}\sum_{i=1}^{m} [\log\left(1 - D(G(x_i))\right)]. \tag{51}$$

The adversarial training strategy aims to maximize L_D and minimize L_G simultaneously. During the solution optimization phase, this framework minimizes the overall loss L by minimizing the reconstruction and generator losses and maximizing the discriminator accuracy, as follows:

$$L = -\lambda_D L_D + \lambda_G L_G + \lambda_{AE} L_{AE}. \tag{52}$$

During the training phase, the framework was optimized using stochastic gradient descent. The autoencoder is updated as follows:

$$\nabla_{\theta_{AE}} \left\|(x_i - \hat{x}_i)\right\|_2^2. \tag{53}$$

The generator is updated as follows:

$$\nabla_{\theta_{AE}} \frac{1}{m}\sum_{i=1}^{m} [\log(1 - D(G(x_i)))]. \tag{54}$$

The discriminator is updated as follows:

$$-\nabla_{\theta_{AE}} \frac{1}{m}\sum_{i=1}^{m} [\log D\left(s_i\right) + \log\left(1 - D\left(G(x_i)\right)\right)]. \tag{55}$$

After the above stages, the trained model can generate mobile user profiles.

The field of the GANS has attracted the attention of other researchers. Li et al. [95] proposed a seedless graph de-anonymization method. The authors used a deep neural network to learn the features, a graph autoencoder to obtain a latent representation, and an adversarial learning model to transform the embedding of anonymous graphs into a latent space of auxiliary graph embeddings. Finally, the model considers the most similar node to be the anchor node and propagates it to the remaining nodes. Pan et al. [96] proposed an adversarial regularized graph embedding framework for community detection. The framework uses a GCN as the encoder, where topological information and node content are embedded into vector representations, graph decoders are built to reconstruct input graphs, and adversarial training principles are applied to force latent codes to match prior Gaussian distributions or to make the codes evenly distributed.

GAN performs well in predicting dynamic user preferences on social networks, and has been widely used in graph-based user profiling [97]. One of the limitations of GAN-based technology is that GAN may not converge. Furthermore, the quality of the samples generated by GAN must be improved [98].

4) FILTERING-BASED APPROACH

Filtering-based techniques can help analyze user information, identify user interests, and recommend suitable content [99]. These techniques primarily include rule-based, content-based, collaborative, and hybrid filtering approach [15].

a: RULE-BASED FILTERING APPROACH

Rule-based filtering was a relatively early approach. The information system first generates rules corresponding to a user according to the information or demographic data filled in during registration. It then matches the user's needs with the rules for recommendation [15]. Choi and Han [100] designed a rule-based filtering system to meet the personalized needs of users. The system first generates an ontology of domain-specific knowledge concepts according to the registration information of the user. The ontology server retains the definitions of domain-specific metadata, principles of data classification, and data correlation. When a user queries, the system's matching engine uses a matching algorithm to check and reflect the user's request more accurately; this is a QoS quality evaluation technique. The system uses a rule-based search engine to infer and test the rules that users query. In addition, the system can reflect user preferences through agents to provide users with personalized services. The system architecture is shown in Fig. 5.

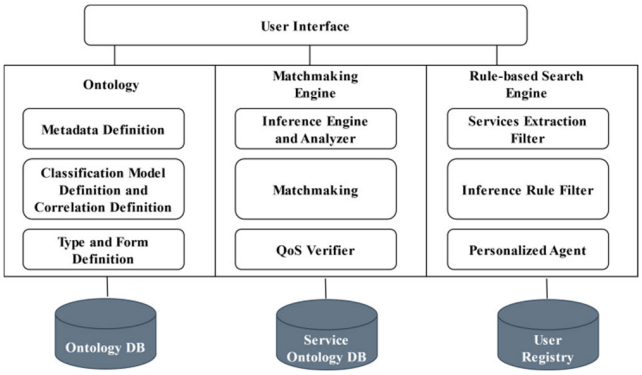

FIGURE 5. Rule-based filtering system architecture [100].

Although the rule-based filtering technology can rapidly derive user profiles without requiring large quantities of sample data, it has several limitations. These rules, which require human configuration, rely on the users' self-descriptions or topics of interest. Consequently, they are difficult to maintain and are prone to bias [100].

b: CONTENT-BASED FILTERING APPROACH

The content-based filtering approach is based on a user's past selections. This approach assumes that users exhibit the same specific behavior in the same situation [101], and recommends content similar to that of the user [102].

Reddy et al. [103] introduced a content-based filtering system for movie recommendation. This system was used as an example to illustrate this logic. The system should recommend similar movies based on the high user ratings. The dataset was divided into two parts: one contained a list of movies and their genres, and the other contained a list of movie ratings. Users rated the content on a scale of 1 to 5, with 5 being the best score. The detailed steps of the algorithm are presented in Algorithm 3.

The Euclidean distance formulated in (2) is often used to measure the straight-line distance between two points [104].

Content-based filtering approaches can also be combined with other algorithms. Shu et al. [105] proposed an approach that combines CNN and content-based filtering. In this approach, based on the historical scores between the students and learning resources, a CNN is used to analyze the potential features from the text information of multimedia resources. It then predicts the scores between students and new learning resources through content-based filtering methods, and selects suitable learning resources to be recommended.

The content-based filtering approach generates recommendations by analyzing a user's project properties and profiles, without relying on other users' information. It has sufficient information to avoid cold starts, making its recommendations easy to interpret. The fact remains, though, that it has several limitations. The diversity and novelty of its recommendation results are insufficient; they can have inaccurate attribute use when selecting items and require a large amount of domain knowledge and sufficient attribute information [99].

Algorithm 3 Content-Based Movie Recommendation

Input: a movie list $l_q = \{l_1, l_2, \ldots, l_q\}$, where l_i includes each movie's ID and genres; a genre matrix $J_{m,n}$, where m is the movie ID, and n is the genre; and a ranking list $Rl_r = \{Rl_1, Rl_2, \ldots, Rl_r\}$, where Rl_i includes each movie's ID and rating number.

1: **for** each matrix item $J_{a,b} \epsilon J_{m,n}$ **do**
2: **if** $(\exists(l_i \in l_q)(l_i.movieID = a \wedge l_i.genres = b)$ **then**
3: $J_{a,b} \leftarrow 1$
4: **end if**
5: **end for**
6: Build a scoring matrix $S_{m,1}$, where m is movie ID
7: **for** each item $S_{a,1} \epsilon S_{m,1}$ **do**
8: **if** $(\exists(Rl_i \in Rl_r)(Rl_i.movieID = a \wedge Rl_i.number > 3)$ **then**
9: $S_{a,1} \leftarrow 1$
10: **else**
11: $S_{a,1} \leftarrow 0$
12: **end if**
13: **end for**
14: Build a result matrix $R_{m,n} = J_{m,n} TS_{m,1}$
15: Convert the result matrix to binary format
16: Calculate the Euclidean distance between the current user and other users
17: Keep the row with the smallest distance as a recommended movie

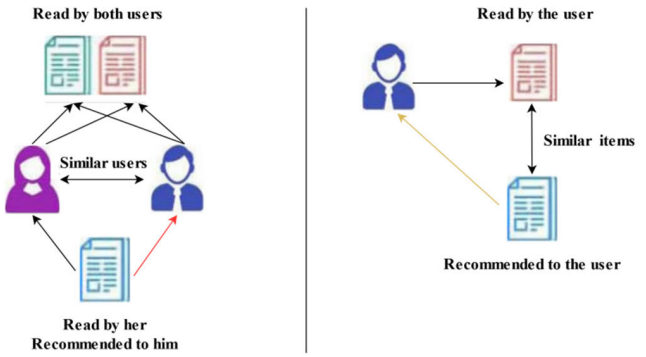

FIGURE 6. User-based and item-based collaborative filtering [107].

c: COLLABORATIVE FILTERING APPROACH

Collaborative filtering uses a sequence of user-item similarity numbers to provide personalized content [106]. Collaborative filtering involves two methods: user- and item-based. An example provided by Wu et al. [107] was used to illustrate this logic. User-based collaborative filtering focuses on the similarity between users. Assuming that multiple users have ratings for books, each user should find neighbor users according to predefined specifications and then calculate the correlation between user ratings. If the ratings of the two users are highly similar, their favorite books are also similar, and mutual book recommendations can be made among them. Item-based collaborative filtering focuses on similarities between item users. For books with similar audiences,

books that a particular audience user has not rated can be recommended. An example is shown in Fig. 6.

Collaborative filtering can be combined with other modules to create a recommendation system. Nilashi et al. [108] proposed a movie recommendation method based on collaborative filtering. There are several stages in model manipulation. First, the expectation maximization (EM) algorithm was used to cluster the users' movie ratings, and the similarity matrix of items and users was generated using SVD dimensionality reduction. Calculations are then performed based on past ratings to determine the similarity between the target user and the other users. Through offline training, the model realizes predictions and recommendations for the target users. The binary Jaccard similarity coefficient was used to calculate the similarity between the two items. A coefficient has a taxonomy x with m categories into which items may fall. Suppose that each item can be represented as a binary vector, $E = \{e_{x,1}, e_{x,2}, \ldots, e_{x,m}\}$. Then, a binary variable $e_{x,p}$ $(p = 1, 2, \ldots, m)$ can be defined as:

$$e_{x,p} = \begin{cases} 1, \text{if x} \in p \\ 0, \text{otherwise} \end{cases} \tag{56}$$

Then, the semantic similarity of the two movies, x and y, can be formulated as:

$$\text{semsim}\,(x, y) = \frac{k_{11}}{k_{01} + k_{10} + k_{11}} \tag{57}$$

where k_{01}, k_{10}, and k_{11} represent the total number of each type for $(e_{xj} = 0; e_{yj} = 1)$, $(e_{xj} = 1; e_{yj} = 0)$, and $(e_{xj} = 1; e_{yj} = 1)$, respectively. The framework of the model is illustrated in Fig. 7.

Su et al. [109] designed a position prediction framework. The framework uses the check-in method to collect users' historical trajectories, and a collaborative filtering method to collect users' implicit preferences. Thus, the framework simulates the influence of multiple social circles. Finally, the framework evaluates the location prediction framework using real datasets. Peng et al. [110] introduced a collaborative filtering model using ideal user groups that dynamically change according to demographic data distribution as dynamic labels. The model considers the popularity of items among different user types and provides recommendations to users with similar demographic characteristics.

The collaborative filtering technique exploits user similarities [79], understands changes in user behavior over time, and produces diverse and incidental personalized lists. It performs well in large user spaces [99]. Nevertheless, when encountering insufficient historical interactions of users, such as dealing with inactive users, collaborative filtering encounters the obstacle of data sparseness [111], which is called the cold-start problem. If the data dimension is exceptionally high, the computational cost of the collaborative filtering system is enormous [112]. In reality, many datasets are sparse, which may prevent collaborative filtering systems from generating accurate recommendations [113].

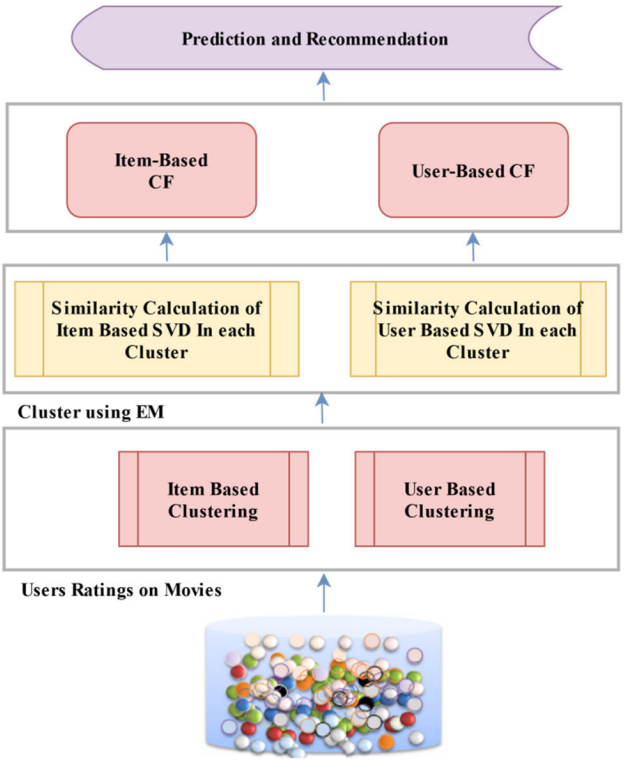

FIGURE 7. Model framework with collaborative filtering involved [108].

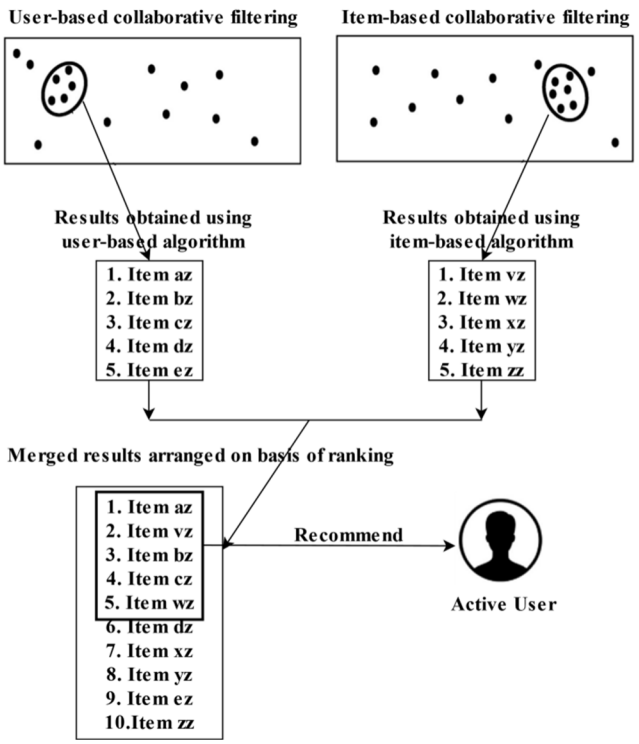

FIGURE 8. Hybrid filtering structure [99].

d: HYBRID FILTERING APPROACH

Hybrid filtering [114] combines different filtering models, such as content-based or collaboration-based models. They also combine one or more technologies [115]. Hybrid

filtering has been used to overcome bottlenecks in collaborative filtering systems, because combining both approaches can prevent inherent pitfalls [15]. Fig. 8 shows an example of a hybrid filtering system. Combining the user- and item-based methods in this example forms a hybrid model.

Hybrid filtering combines the advantages of different filtering technologies and overcomes the main shortcomings of content- and collaboration-based systems, such as cold start, sparsity, and gray-sheep problems. One of its limitations is the high implementation cost. There is a high degree of complexity in terms of time and space. Moreover, considering privacy issues, it is challenging to collect explicit information [99].

5) KNOWLEDGE-GRAPH-BASED APPROACH

A knowledge graph is a directed information network in which nodes and edges represent entities and relationships, respectively. [116]. A knowledge graph contains rich information that can be integrated with user behavior data, expand hidden relationships, and more accurately identify user profiles [117]. Rich information in a knowledge graph can also help solve critical problems in user profiling and recommendations, such as data sparsity, cold starts, and recommendation diversity [118].

Fig. 9 shows an example of user profiling based on a knowledge graph. The entities in the figure include the user, movie, actor, director, and type, and the relationships between entities include interaction, belonging, performance, director, and friendship. Based on the knowledge graph, movies and users can be connected through potential relationships, which helps improve the recommendation performance.

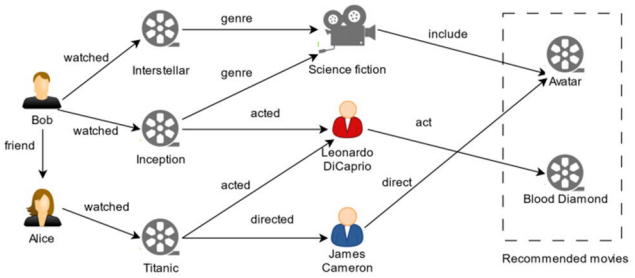

FIGURE 9. User profiling based on knowledge graph [118].

The knowledge graph has the following representation: $G = (E, R, S)$, where E represents the entities $\{E_1, E_2, \ldots, E_n\}$, R represents all relationships $\{R_1, R_2, \ldots, R_n\}$, and S represents triples $\{S_1, S_2, \ldots, S_n\}$. Each triple S_i has three parts: head entity, relationship, and tail entity.

Knowledge graph methods include embedding, connection, and propagation-based methods, which are introduced below.

a: EMBEDDING-BASED APPROACH

Embedding-based methods utilize graph-embedding modules to learn entities and relationships in knowledge graphs, where rich facts in knowledge graphs can be used to enhance the representation of items or users. This type of method represents entities and relationships through graph embedding methods, and then expands the semantic information. Zhang et al. [117] proposed a learning method for embedding heterogeneous entities based on a knowledge-based representation to analyze and recommend Amazon's e-commerce dataset, achieving better results than other baselines.

First, we encode the knowledge of users and projects in a graph structure. Specifically, the user-item knowledge graph consists of a set of triple structures, each consisting of a head entity i, tail entity j, and a relationship from i to j. For example, "buy" indicates that the user has purchased this product. Other relationships include "belong_to_category," "belong_to_brand," "mention word," "also_view." The authors then projected each entity and its relationship onto a single low-dimensional embedding space. For example, for the triplet (i, j, r), the embedding vectors are e_i, e_j, e_r. According to the graph-embedding method, the embeddings of the head and tail entities should be similar. In other words, it is expected that $\text{TRANS}_{er} \approx e_j$. Considering all triples S, the following loss function can be defined to learn the embeddings:

$$L = \sum_{(i,j,r)\in S} \left\{ \sum_{(i,j',r)\in S^t} \left[\gamma + d\left(\text{TRANS}_{e_r}(e_i), e_j\right) - d\left(\text{TRANS}_{e_r}(e_i), e_{j'}\right)\right] + \sum_{(i',j,r)\in S^h} \left[\gamma + d\left(\text{TRANS}_{e_r}(e_i), e_j\right) - d\left(\text{TRANS}_{e_r}(e_{i'}), e_{j'}\right)\right] \right\} \tag{58}$$

where S^t is a set of negative triples with random entities replacing the tail, S^h is another set of negative triples with random entities replacing the head, $d(.)$ is a metric function that measures the distance between two embeddings, and $\text{TRANS}_{e_r}(e_i)$ is a transformation function or even a neural network. The model can be learned using stochastic gradient descent (SGD).

When the above model is optimized, the entity and relationship embeddings are obtained and used for recommendations. For example, assume that the embedding of the *buy* relationship is e_{buy}, and that the target user of the embedding is e_u. Then, the candidate item e_j can be sorted according to the distance $d(\text{TRANS}_{e_{buy}}(e_i), e_j)$ between the candidate items e_j and e_u and recommended.

Other scholars have also conducted research in this field. Zhu et al. [119] proposed a method that combined knowledge graph embedding and collaborative filtering. By maintaining the original structure and semantic information, the author first learns the entities and relationships in the graph through the knowledge graph embedding method, and then integrates the semantic information into collaborative filtering through semantic similarity calculation between items.

This embedding-based method is flexible and suitable for most scenarios. However, it is difficult to interpret and capture high-order relationships [118].

b: CONNECTION-BASED APPROACH

Connection-based methods support item recommendations by mining multiple connection relationships between the users and items. A heterogeneous information network formed based on user-project interaction data, the meta-path method, can be introduced in the recommendation system, where heterogeneous information networks and meta-paths are defined as follows:

Heterogeneous information network: Take a directed graph $G = (V, E)$, including mapping function of node type $\emptyset : V \to A$ and link type $\emptyset : E \to B$, where for any node $V_i \in V$ exists a specific node type $\emptyset(V_i) \in A$, and each link $E_j \in E$ belongs to a specific relationship type $\emptyset(E_j) \in R$. If there is a node type number $|A| > 1$ or a link type number $|B| > 1$, it is called a heterogeneous information network.

Meta-path: This path comprises a series of relationship sequences between the different types of nodes. Its formal definition is $P = A_0 \to A_1 \to \cdots \to A_k$, where A_0 to A_k are node types defined in network G and meta-path P is a composite relationship $R_1 R_2 \ldots R_k$ from A_0 to A_k.

Researchers can use the PathSim function as the connectivity similarity evaluation method [120], which is defined as

$$s(x, y) = \frac{2 \times |\{P_{x\sim y} : P_{e\sim e_j} \in P\}|}{|\{P_{x\sim x} : P_{x\sim x} \in P\}| + |\{P_{y\sim y} : P_{y\sim y} \in P\}|} \quad (59)$$

where P is a symmetric meta-path, x and y are two objects of the same type, and $P_{m\sim n}$ is the path instance between m and n.

The following are the research cases of some researchers: Yu et al. [121] extracted the meta-path in the shape of user-item -*-item in the knowledge graph, calculated the feature value between users and items based on user preference according to the PathSim method, extended the feature value calculation to all users and items to obtain the users' preference feature matrix, decomposed this feature matrix through the matrix decomposition method to obtain the feature vector of each meta-path, and finally calculated the relationship between all users and items under different meta-paths. The recommended result was obtained from the dot-product weighted sum of the feature vectors. Compared with meta-paths, meta-graphs can depict complex feature information in heterogeneous information networks.

Zhao et al. [122] used meta-graphs to perform feature extraction in information networks. Standard matrix factorization (MF) was performed for each meta-graph-generated similarity to generate the latent features. They then used a factorization machine (FM) to automatically learn from the observed ratings and quickly select useful meta-graph-based features. Sun et al. [123] analyzed the semantic paths between users and items. Each semantic path was modeled using an RNN to obtain a representation of each semantic path. Subsequently, a pooling layer-based method was used to obtain the characteristics of the overall semantic path from the user to the item. Finally, a fully connected network was used to predict the recommendation results.

The connection-based method is more interpretable; however, information will inevitably be lost if complex user-item connection patterns are deconstructed into separate linear paths. In addition, this method does not adequately support sparse datasets because it obtains sufficient paths in user profiling scenarios [118].

c: PROPAGATION-BASED APPROACH

To fully utilize the information in knowledge graphs, some researchers have proposed propagation-based methods that combine the representation of entities and relationships with high-order connection patterns to achieve more personalized recommendations. The commonly used implementation of propagation-based methods aims to refine entity representations by aggregating neighbor embeddings in knowledge graphs, and then using rich representations of users and potential items to predict user preferences.

Wang et al. [116] proposed a framework for integrating knowledge graphs into recommender systems, called RippleNet. to introduce a preference propagation mechanism into a knowledge graph. It assigns weights to neighbors in the graph by training a relationship matrix $R_i \in R^{d\times d}$ and performs aggregation operations in each layer of the triple set. First, they used the head entity $e_{hi} \in R^d$, relationship matrix R_i, and candidate $v_i \in R^d$ to calculate the weight P_i of the tail entity in the corresponding triplet using the following formula: The similarity of candidate item v_j to the neighbor of the interacting item was calculated. Second, they used the weighted average of the tail entity embeddings to calculate the user representation at the h-th level of the triplet set using the following formula:

$$p_i^h = \frac{\exp\left(v_j^T R_i^h e_{h_i}^h\right)}{\sum_{(e_{h_k}, r_k, e_{i_k}) \in S_{u_i}^h} \exp\left(v_j^T R_k^h e_{h_k}^h\right)} \quad (60)$$

$$o_{u_i}^h = \sum_{(e_{h_i}, r_i, e_{t_i}) \in S_{u_i}^h} p_i^h e_{t_i}^h \quad (61)$$

The candidate items embedded in the first layer of the triplet set were initialized with $v_j^{initial}$ and replaced with $o_{u_i}^{h-1}$. Then, by repeating (60)–(61), the framework can propagate the user preference from interactive items to distant neighbors. Finally, the user preference score was calculated using the following formula:

$$\hat{y}_{i,j} = \sigma\left(u_i^T v_j^{\text{initial}}\right), \quad (62)$$

where $\sigma(x)$ is the sigmoid function.

Other researchers have explored user-project propagation mechanisms. For Instance, Wen et al. [124] proposed a neighborhood interaction model that further considered the interaction between item- and user-side neighbors. After obtaining a high-order representation of entities in the KG,

the model uses the enhanced representation of the user and item neighbors to predict user preferences.

The propagation-based method, which is also interpretable and has more advantages regarding the depth of information mining in the knowledge graph, is unsuitable for the user-profiling scenarios of large-scale datasets [118].

E. PERFORMANCE EVALUATION

To evaluate modeling performance, researchers have used metrics to calculate the results. The following metrics are commonly used [15], [21].

Parameter Definition

TP: Number of samples that correctly predict whether a user belongs to a particular profile.

TN: Number of samples that correctly predict that a user does not belong to a particular profile.

FP: Number of samples that incorrectly predict whether a user belongs to a particular profile.

FN: Number of samples that incorrectly predict that a user does not belong to a particular profile.

Accuracy (ACC): This refers to the percentage of correct predictions (both positive and negative), defined as:

$$\text{Accuracy (ACC)} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{63}$$

Precision (PRE): This refers to the number of correct positive predictions in proportion to all the positive predictions. This indicates how many positive samples among all the predicted samples are predicted correctly, which is defined as:

$$\text{Precision (PRE)} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{64}$$

Recall (REC): this refers to the correct positive predictions in proportion to the true number of positives. This indicates how many positive samples among the total samples are predicted correctly, which is defined as:

$$\text{Recall (REC)} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{65}$$

F1: Both precision and recall are considered, which are defined as

$$\text{Recall (REC)} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{66}$$

Specificity: This refers to correcting the negative predictions in proportion to the true number of negatives. This is also called the true negative rate (TNR), which is defined as

$$\text{Specificity} = \frac{\text{TN}}{\text{FP} + \text{TN}} \tag{67}$$

Hit ratio: This refers to the ratio of the correct prediction number to the target item number, which is defined as:

$$\text{HitRate} = \frac{\text{Number of hits}}{n} \tag{68}$$

Click-through Rate (CTR): This is the ratio of the total number of clicks on an item to the number of times the item is shown to users. It is defined as

$$\text{CTR} = \frac{\text{Number of clicks}}{n_t} * 100 \tag{69}$$

V. INDUSTRIAL APPLICATIONS

Social-network-based user profiling technology has revolutionized the manner in which industries analyze and manage users. By leveraging the vast amount of data available on these platforms, businesses can gain valuable insights into their customers' preferences and behaviors, enabling them to provide tailored recommendations and experiences. Table 8 lists leading industrial applications.

A. GAME

Game platforms leverage user profiling technology to predict users' personalities, preferences, and interests. This approach is used to design personalized game scenes or mechanisms to improve game user participation and stickiness. The datasets used for research in this area primarily come from games, including user behavior, attributes, and social networks, combined with user questionnaires to collect user opinions on games [124]. Various analytical techniques, such as correlation analysis, regression analysis, and feedforward neural networks, have been used to study these data [125]. Autoencoders are also used to take full advantage of label correlations without destroying training-test consistency. RNN-type models, such as LSTM, are used for scenarios in which the time factors must be considered. The graph-based model GCN [126] analyzes multisource features and multirelationship graphs.

B. IMAGE AND SHORT VIDEO PLATFORM

Currently, image and short video platforms such as Instagram, Facebook, TikTok, Twitter, and Musical are widely used. These platforms provide rich pictures and videos posted by people that can be used as data sources for user profiling. Several studies have combined pictures, videos, and captions for joint analysis [31], and CNN models, including LeNet, VGGNet, AlexNet, ResNet, Inception-v3, and GooglNet, have been used for image modeling [127], [128]. Some studies designed independent models for different user profiles, whereas others merged the models and used different classifiers [76]. Increasing the accuracy of model training requires standardizing image specifications and horizontal mirror image enhancement techniques before training. Video processing can be converted into image processing by extracting the main frames of the short videos. The hierarchical multihead attention mechanism can effectively learn video statistical information and user-adjusted dynamic features [127]. The results of user profiling include the gender, age, country, personality, interests, and user preferences.

TABLE 8. Industrial application analysis.

Industries	Application Description	Data Collection	Techniques and Algorithms
Game	It predicts the user's personality, preferences, and interests through game settings, user archives, user questionnaires, and other data to design personalized game scenes or mechanisms in a targeted manner [124][125][126]	User behavior, attributes in games, social networks, and user-oriented questionnaires	BiLSTM, R-GCNs, regression analysis, correlation analysis, and feedforward neural network
Image and short video platform	It predicts user profiles such as gender, age, country, personality, interests, and preferences by analyzing pictures and short videos on the social network [76][127][128]	Pictures, videos, and related text posted by people on social networks	CNN, image normalization and enhancement, and attention mechanism
Education	It predicts students' learning interests and preferences and recommends appropriate educational resources through student learning data on the course platform and student behavior data on social networks [5][35][129]	Student class data on the classroom platform, student information on social networks, and questionnaire data for student users	XGB regression analysis, random forest, SVM, LightGBM, and content-based/rule-based filtering
Recommendations for work and life	It predicts users' interests and preferences through user information and activity data on social networks and recommends suitable resources, such as job opportunities and news [34][41][130][131]	User basic information and user behavior data on social networks and candidate jobs or news resources in the resource library	TF-IDF, LDA, user ontology, cosine similarity, K-NN, content-based/rule-based filtering, GNN, CNN, BiLSTM, and attention mechanism
Personalized Q&A	It generates text responses that fit the user's Q&A context based on the user's historical conversation data on social networks [30][132][133][134]	User historical conversation data on social networks such as Weibo and Reddit	GNN, MLP, CNN, and GRU
Online shopping	It predicts users' purchasing intentions based on their purchasing behavior on online shopping platforms, such as items, quantities, and purchasing preferences [135][136][137]	User purchase and activity data, including purchased items, prices, user ratings, comments, and timestamps	Cosine similarity, Jaccard similarity, content-based filtering, CNN, LSTN, and self-attention mechanism
Location prediction	It predicts the user's location based on user behavior data on social networks, such as posts, comments, check-ins, and friend influence, and makes destination recommendations [138][139][140]	Comments, business data, user check-in data, and friend influence on social networks	LDA, doc2vec, cosine similarity, Jaccard similarity, GNN, multi-head attention mechanism, and graph-based method

C. EDUCATION

User profiling is widely used in the education industry to predict students' learning effects or to recommend appropriate courses based on their behavioral characteristics. In education, data sources include student class data captured from course platforms, including student interactions, preferences, cognitive abilities, and learning status. Student information, such as age, region, and education, is also captured from social networks, such as Facebook or LinkedIn. Furthermore, other data sources, such as student surveys containing background, interest, student-type test data, intelligence, and ability test data, can act as data sources [35]. Feature transformers can be used to obtain student features. Ontology technology can store the user characteristics. The pipeline contains candidate steps, XGB regression, random forest, SVM, and LightGBM algorithms, which can be used to train and test the data. Rule-based and content-filtering techniques can recommend the educational resources that students need [5], [129].

D. RECOMMENDATIONS FOR WORK AND LIFE

People offer many recommendations in work and life, such as jobs and reading news. In this regard, user profiling provides an excellent support. User profiling is an integral part of providing recommendations for work and life. By analyzing a user's basic information and historical behavior, user profiling can provide suitable recommendations for job opportunities or news that a user may be interested in [41]. Datasets used for user profiling mainly contain basic user information, user behavior data, candidate jobs, and news resource libraries. User information includes gender, age, region, language, educational background, work experience, skills, and personal expectations. User behavior data include user search information, collection information, clicks, ignores, ratings, likes, follows, and comments on social networks [130]. With regard to technology, TF-IDF and LDA can be used to evaluate words and topics. Ontology technology is used to record features of users' information and target items, such as positions and news. Cosine

similarity and K-NN are used in content-based user classification, and rule-based filtering techniques are recommended [131]. Artificial neural network technologies such as GNN, CNN, and RNN have been used to model massive and complex datasets [34].

E. PERSONALIZED QUESTIONS AND ANSWERS

Today, personalized question answering has become increasingly popular in various fields, including automatic customer services and voice assistants. Personalized questions can bridge the gap between available resources and the number of people, mitigating risks associated with social stigma and eliminating the fear of being judged by others [132]. In this field, user profiling technology can analyze a user's language style and preferences based on historical conversations on social networking sites, such as Weibo and Reddit. This technology can generate responses suitable for users based on the context [30]. The data source was the historical question-and-answer texts of users on social networks. Because the context of the statement is essential, the subject model often uses an RNN, which can use the user's historical questions and answers. Therefore, deep learning technologies, such as CNN, MLP, and GRU, are used in a specific process [133], [134].

F. ONLINE SHOPPING

Online shopping platforms, such as Amazon and Taobao, use user profiling to provide personalized customer recommendations. By analyzing a user's shopping behavior, including purchased items, categories, prices, user ratings, comments, and timestamps, these platforms can derive a user's shopping preferences and suggest products accordingly. Moreover, recommendations for users' purchasing behavior can be provided, ultimately increasing customer loyalty and sales. In this field, the data source for user profiling is user transaction and activity data from online shopping platforms, including purchased items, categories, prices, user ratings, comments, and timestamps [135]. For item recommendation, some studies used content-based filtering technology, which uses the cosine similarity method to test the relationship between users and items, providing users with a list of recommended items [136]. Some studies have used logistic regression and CNN to predict users' purchase intentions and ratings. LSTN, CNN, and attention mechanisms are used for application scenarios that must predict short- and long-term user purchase intentions. Therefore, a CNN is used to capture short-term needs. Moreover, the self-attention mechanism captures long-term cyclical needs and ultimately generates recommendations [135], [137].

G. LOCATION PREDICTION

In recent years, there has been an increase in the use of location-based social networks (LBSNs), which combine satellite positioning with social network information. These networks allow users to share their location and access experiences, and provide valuable information for user profiling systems. Using LBSN information, a user profiling system can analyze and predict the user behavior. In this field [138], the primary data sources for user profiling include comments on social networking sites, such as Twitter, business data, user check-in data, and friend influence. LDA technology can obtain user interest features from comments. Moreover, the doc2vec model can perform word embedding encoding and cosine similarity. Jaccard coefficient similarity algorithms can be used to calculate the similarity of user feature vectors. In addition, the user characteristics aware recommendation algorithm (ISC-CF) can comprehensively evaluate the impact of user interest information, social relationships, and geographical location [139]. As a result, some studies have used graph-based models to solve the sparse data problem. The model associates the user, application, and location information, and generates high-order features through graph propagation and aggregation. [140].

To sum up, user profiling technology based on social networks is widely used in various industries, and the data collection methods and calculation algorithms are also formulated according to the characteristics of different industries.

VI. ETHICAL AND LEGAL ASPECTS

Although the technical aspects of user profiling are essential, technology is human-centered, so ethical and legal elements are indispensable. We discuss this aspect below.

A. FAIRNESS AND BIAS

User profiling uses specific algorithms to predict user behavior and provide recommendations. However, if these algorithms are biased, fairness of the results cannot be guaranteed. For example, the fairness of rankings may be affected by position bias. Therefore, topics with high rankings are more likely to receive attention and improve rankings, which is the opposite of the case for low-ranking topics. Poor working environments affect the fairness of passenger satisfaction on ride-hailing platforms, leading to discrimination against ethnic underrepresented groups [141]. Automated recruitment systems match positions, and candidates will be affected by bias and discrimination based on age, gender, and race. In this regard, bias and fairness issues are morally unreasonable and violate the relevant safeguards of fair law [142]. Based on this algorithm, the unfairness and discrimination continue to increase. Therefore, models are required to quantify and solve these issues. In addition, the data source significantly affects the prediction results of user profiling if many false data points are published. For example, the inclusion of false-negative feedback in a specific project will seriously affect fairness [143]. Therefore, targeted algorithms that can distinguish between true and false information must be developed to prevent the introduction of this type of bogus data, which can affect the system.

B. DATA OWNERSHIP AND CONTROL

The user profiling dataset was derived from various user data. As data owners, users should decide which data to

provide and how to use them. However, data generated on the internet, including social media-generated comments or user comments and photos posted by users, are collected and used without the consent of users; this is called data serfdom [144]. Data abuse is considered unethical. Therefore, much work must be done to protect the rights and data of internet users. Technical methods can also play a significant role in preventing the misuse of user data. For example, fine-grained user control over geographic information sovereignty can be added with decentralized data storage and a discrete global grid system [145]. Using blockchain technology, users can consider data control, ownership, and sharing by verifying the object, time, content, method, and purpose of data sharing [146].

C. PRIVACY PROTECTION

Although user profiling technology has brought much help and convenience, there are also hidden dangers to user privacy protection. For example, after a user registers on a social website, an attacker can use personal information and user activity for malicious activities [147]. Although social networks are popular because people can share topics of interest, leaks or abuse of user information can lead to serious personal privacy risks and consequences [148]. Unfortunately, individual posts on these social networking sites may be used maliciously, indirectly revealing user privacy through user profiling technology. In particular, if users have accounts on multiple social networks and posts, these cross-platform user profiles and social behaviors may be used to infer the user's identity more accurately [149]. Once hidden initial personal information on the social network is leaked, it may affect the user's public image. Career development may even lead to severe consequences such as tracking, identity discrimination, and theft [150]. Therefore, people have a reason to worry about the moral and legal risks of user profiling technology. Therefore, further protection of personal privacy is required.

D. LEGAL COMPLIANCE

User profiling must comply with relevant laws and regulations, such as the General Data Protection Regulation (GDPR), for various industrial applications that are helpful to humans. For example, adult education entities must obtain consent according to the GDPR requirements in online education. Moreover, teachers must provide consent before their names, images, and voices can be used or recorded. The passwords must be changed regularly. Access to shared credentials must be disabled, and steps must be taken to protect devices and profiles where user information is stored [151]. Therefore, it is necessary to develop research models and frameworks to ensure legal compliance. For example, in the geographic and social network (GeoSN) field, risk models can help understand users' threat risks in the spatial, socio-semantic, and time dimensions, helping online social network platforms incorporate risk prevention into their design [152]. Technical methods are also used to promote compliance with relevant laws. Developing a system based on a semantic framework can generate a semantic framework representation for the data processing agreement (DPA) text and compare it with GDPR to check the data DPA and GDPR [153].

E. ACCOUNTABILITY AND OVERSIGHT

As an artificial intelligence technology, user profiling prediction and recommendation results will specifically affect users. Therefore, accountability and oversight mechanisms are needed to ensure that its practices are ethical and legal. Accountability includes authoritative recognition, questioning, and limiting authority. However, the multiple characteristics of accountability relationships must be considered. Accountability goals include compliance, reporting, monitoring, and enforcement [154], and some frameworks and standards have been attempted. Improving machine learning for automatic and algorithmic decision-making (ADM) is an evaluability framework that decomposes administrative management processes into technical and organizational elements and is used to determine contextual record-keeping mechanisms, thereby providing a purposeful review of decisions and processes [155]. As a new software quality standard, legal responsibility reflects the degree to which software is responsible for the law. This requires a corresponding requirements analysis and system design. Its primary attributes are traceability, integrity, suitability, effectiveness, and continuity. Therefore, strengthening the requirements for legal liability can reduce the cost of software being legally responsible [156].

VII. FUTURE DIRECTIONS

Based on the current research status of user profiling based on social networks, this section provides the following future research directions.

A. USER PROFILING WITH NON-TEXT DATA

Since research on user profiling based on text data began, several achievements have been made in this field. Today, people appear to be particularly inclined to publish non-text data on Twitter, Facebook, and Instagram [16]. Therefore, there is considerable potential to conduct research in this field. Progress has been made in this regard. For example, Biswas et al. [157] combined feature annotations of text and images on Twitter, extracted image tags using the Google Cloud Vision API, and classified Twitter images into five large personality traits using a fully connected neural network. However, converting images into text labels without using deep features involves text processing. Alamdari et al. [158] used five CNN models to extract the features of the product images. Subsequently, they calculated image similarity and made recommendations based on these features. In contrast, the dataset used in this study is a well-known movie dataset called Movielens, which is relatively topic-specific and easy to process. Therefore, dealing with other datasets may be challenging. Strukova et al. [159]

developed a framework to download a photography dataset that contained pictures, social user comments, and career information from Flickr to predict whether the user was a professional photographer. After preprocessing and feature extraction, the photography dataset included pictures and user comment features. Machine-learning algorithms were then used to complete the prediction. Nevertheless, whether the relatively prominent features extracted from images are more effective than those obtained through deep-learning methods is worth further exploration [160].

B. DYNAMIC USER PROFILES

Most studies have focused on static user profiles that are difficult to change, such as predicting the gender, age, and personality of the user. In contrast, dynamic user profiles, such as interests and preferences, change over time and can provide more targeted predictions or recommendations [22]. Therefore, further research in this field will be valuable, and if the recommended content on a social network can follow a user's interests, it will significantly increase their interest in participation. Several studies on this topic have been conducted. Cheng et al. [161] proposed a personalization mechanism for learning users based on their long- and short-term behavior profiles to improve both document reranking (DR) and next query prediction (NQP) performance in ongoing search sessions. By contrast, this approach requires external knowledge of the model, which restricts its performance. Reyes et al. [44] proposed a clothing recommendation system based on user preferences, using two methods to recommend short- and long-term products. Short-term recommendations are continuously updated with user interactions and long-term offers are updated at intervals. Conversely, this study only considered clothing data sources; therefore, this approach should be validated using more significant data.

C. ACROSS DIFFERENT SOCIAL NETWORK

Some user-profiling studies have been based on a single platform. In contrast, more users participate in multiple social networks, such as Twitter, Facebook, Instagram, and TikTok, to consume more content [162]. Different social networks provide unique services such as Facebook for making friends, LinkedIn for making work connections, Twitter for finding information, and YouTube for sharing videos. Thus, people tend to use multiple social networks [163]. Therefore, cross-platform user profiling is a promising research topic. Several studies on this topic have been conducted. Zhou and Yang [164] proposed a user account matching method based on location verification that uses the latitude, longitude, and time coordinates of the user on different platforms to perform user similarity matching to determine the same user. This approach has certain advantages for recording location tags but is challenging if it is based only on social networks, without the help of location tags. Xiao et al. [165] used various mechanisms to infer the possible accounts of users on different social networks, based on the correlation between a mobile device's status and social network events. The drawback of this method is that the cell phone must act as a connecting device and play an indispensable role. Problems may be encountered if users access webpages through social networks.

D. PRIVACY PROTECTION

User personal information deposited on social networks can provide data for user profiling, making it necessary to strengthen privacy protection [166]. Several studies have explored this field of research. For example, Brandão et al. [167] used privacy-preserving clustering to construct profiles. The server calculates the centroid without accessing underlying data. Subsequently, federated learning was used to develop a model for predicting permission decisions in a distributed manner, and all the data were stored on the user's local device. Huo et al. [168] proposed a privacy-preserving global user model based on fuzzy reasoning that protects parameters from untrustworthy ones by allowing participants to train local clusters on the data using the CLIQUE algorithm, and encrypt the clustering parameters using the Paillier method. Although these studies explored privacy protection using special techniques in certain scenarios, more complicated scenarios and environments exist in social networks, highlighting the need for further research in this field.

VIII. CONCLUSION

In this review, we investigated user profiling based on social networks and summarized the recent research in this field. This review introduces user profiling concepts and process phases and elaborates on the main models and techniques. In addition, it illustrates the valuable applications of social network-based user profiling in various industries. Considering the importance of non-technical elements, it also discusses ethical and legal issues from a human-centered perspective. Finally, future research directions are identified, which can help to make future research more precise and valuable. User profiling based on social networks can benefit individuals under various scenarios. Therefore, this topic warrants further investigation.

REFERENCES

[1] S. Rathi, J. P. Verma, R. Jain, A. Nayyar, and N. Thakur, "Psychometric profiling of individuals using Twitter profiles: A psychological natural language processing based approach," *Concurrency Comput., Pract. Exper.*, vol. 34, no. 19, p. e7029, Aug. 2022.
[2] K. G. Gatziolis, N. D. Tselikas, and I. D. Moscholios, "Adaptive user profiling in e-commerce and administration of public services," *Future Internet*, vol. 14, no. 5, p. 144, May 2022.
[3] S. Yan, T. Zhao, and J. Deng, "Interaction-aware hypergraph neural networks for user profiling," in *Proc. IEEE 9th Int. Conf. Data Sci. Adv. Anal. (DSAA)*, Oct. 2022, pp. 1–10.
[4] Z.-H. Deng, C.-D. Wang, L. Huang, J.-H. Lai, and S. Y. Philip, "G^{3}SR: Global graph guided session-based recommendation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 12, pp. 9671–9684, Mar. 2022.
[5] R. Kaur, D. Gupta, M. Madhukar, A. Singh, M. Abdelhaq, R. Alsaqour, J. Breñosa, and N. Goyal, "E-learning environment based intelligent profiling system for enhancing user adaptation," *Electronics*, vol. 11, no. 20, p. 3354, Oct. 2022.

[6] S. Safavi and M. Jalali, "DeePOF: A hybrid approach of deep convolutional neural network and friendship to point-of-interest (POI) recommendation system in location-based social networks," *Concurrency Comput., Pract. Exper.*, vol. 34, no. 15, p. e6981, Jul. 2022.

[7] R. Menaha and V. E. Jayanthi, "Finding experts in community question answering system using trie string matching algorithm with domain knowledge," *IETE J. Res.*, vol. 70, no. 3, pp. 2602–2614, Mar. 2024.

[8] M. Y. Yasar and M. Kaya, "Author-profile-based journal recommendation for a candidate article: Using hybrid semantic similarity and trend analysis," *IEEE Access*, vol. 11, pp. 45826–45837, 2023.

[9] K. Biswas, P. Shivakumara, U. Pal, T. Chakraborti, T. Lu, and M. N. B. Ayub, "Fuzzy and genetic algorithm based approach for classification of personality traits oriented social media images," *Knowl.-Based Syst.*, vol. 241, Apr. 2022, Art. no. 108024.

[10] W. Hong, L. Li, T. Li, and W. Pan, "IHR: An online recruiting system for Xiamen talent service center," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2013, pp. 1177–1185.

[11] P. Jomsri, "Book recommendation system for digital library based on user profiles by using association rule," in *Proc. Int. Conf. Innov. Comput. Technol. (INTECH)*, Aug. 2014, pp. 130–134.

[12] Y.-C. Fan, Y.-C. Chen, K.-C. Tung, K.-C. Wu, and A. L. P. Chen, "A framework for enabling user preference profiling through Wi-Fi logs," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 3, pp. 592–603, Mar. 2016.

[13] M. Amoretti, L. Belli, and F. Zanichelli, "UTravel: Smart mobility with a novel user profiling and recommendation approach," *Pervas. Mobile Comput.*, vol. 38, pp. 474–489, Jul. 2017.

[14] G. Suchacka and J. Iwański, "Identifying legitimate web users and bots with different traffic profiles — An information bottleneck approach," *Knowl.-Based Syst.*, vol. 197, Jun. 2020, Art. no. 105875.

[15] C. I. Eke, A. A. Norman, L. Shuib, and H. F. Nweke, "A survey of user profiling: State-of-the-art, challenges, and solutions," *IEEE Access*, vol. 7, pp. 144907–144924, 2019.

[16] V. R. de Azevedo, N. Nedjah, and L. de Macedo Mourelle, "Identification of client profile using convolutional neural networks," in *Proc. Int. Conf. Comput. Sci. Appl.*, Cagliari, Italy. Cham, Switzerland: Springer, 2020, pp. 103–118.

[17] P. Stefanovic and S. Ramanauskaite, "Travel direction recommendation model based on photos of user social network profile," *IEEE Access*, vol. 11, pp. 28252–28262, 2023.

[18] J. C. Gomez, J. Moreno, M.-A. Ibarra-Manzano, and D.-L. Almanza-Ojeda, "Reconstructive classification for age and gender identification in social networks," *IEEE Trans. Computat. Social Syst.*, vol. 11, no. 2, pp. 2291–2301, May 2023.

[19] S. Nagar, F. A. Barbhuiya, and K. Dey, "Towards more robust hate speech detection: Using social context and user data," *Social Netw. Anal. Mining*, vol. 13, no. 1, p. 47, Mar. 2023.

[20] M. M. Dehshibi, B. Baiani, G. Pons, and D. Masip, "A deep multimodal learning approach to perceive basic needs of humans from Instagram profile," *IEEE Trans. Affect. Comput.*, vol. 14, no. 2, pp. 944–956, Jun. 2021.

[21] S. Kanwal, S. Nawaz, M. K. Malik, and Z. Nawaz, "A review of text-based recommendation systems," *IEEE Access*, vol. 9, pp. 31638–31661, 2021.

[22] G. Piao and J. G. Breslin, "Inferring user interests in microblogging social networks: A survey," *User Model. User-Adapted Interact.*, vol. 28, no. 3, pp. 277–329, Aug. 2018.

[23] S. Ouaftouh, A. Zellou, and A. Idri, "User profile model: A user dimension based classification," in *Proc. 10th Int. Conf. Intell. Syst., Theories Appl. (SITA)*, Oct. 2015, pp. 1–5.

[24] Y. Liu, J. Hou, and W. Zhao, "Deep learning and user consumption trends classification and analysis based on shopping behavior," *J. Organizational End User Comput.*, vol. 36, no. 1, pp. 1–23, Mar. 2024.

[25] S. Alaoui, Y. E. B. E. Idrissi, and R. Ajhoun, "Building rich user profile based on intentional perspective," *Proc. Comput. Sci.*, vol. 73, pp. 342–349, Jan. 2015.

[26] A. Mlaiki, I. Walsh, and M. Kalika, "Why do we continue using social networking sites? The giving loop that feeds computer-mediated social ties," *Systèmes d'Inf. Manage.*, vol. 22, no. 2, pp. 5–47, Sep. 2017.

[27] Y. Liu, Y. Zhang, and X. Zhang, "Level neural model for sequential recommendation," *Discrete Dyn. Nature Soc.*, vol. 2021, pp. 1–12, Jan. 2021.

[28] A. Barforoush, H. Shirazi, and H. Emami, "A new classification framework to evaluate the entity profiling on the web: Past, present and future," *ACM Comput. Surv.*, vol. 50, no. 3, pp. 1–39, May 2018.

[29] R. Norouzi, H. Baziyad, E. Aknondzadeh Noghabi, and A. Albadvi, "Developing tourism users' profiles with data-driven explicit information," *Math. Problems Eng.*, vol. 2022, pp. 1–14, Jun. 2022.

[30] Z. Ma, Z. Dou, Y. Zhu, H. Zhong, and J.-R. Wen, "One chatbot per person: Creating personalized chatbots based on implicit user profiles," in *Proc. 44th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2021, pp. 555–564.

[31] S. Jayarathna, A. Patra, and F. Shipman, "Unified relevance feedback for multi-application user interest modeling," in *Proc. 15th ACM/IEEE-CS Joint Conf. Digit. Libraries*, Jun. 2015, pp. 129–138.

[32] M. Bilal, A. Gani, M. I. U. Lali, M. Marjani, and N. Malik, "Social profiling: A review, taxonomy, and challenges," *Cyberpsychol., Behav., Social Netw.*, vol. 22, no. 7, pp. 433–450, Jul. 2019.

[33] O. Serrat, "Social network analysis," in *Knowledge Solutions: Tools, Methods, and Approaches to Drive Organizational Performance*. Singapore: Springer, 2017, pp. 39–43.

[34] Y. Feng and B. Cautis, "IGNiteR: News recommendation in microblogging applications," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2022, pp. 939–944.

[35] C. K. Pereira, F. Campos, V. Ströele, J. M. N. David, and R. Braga, "BROAD-RSI—Educational recommender system using social networks interactions and linked data," *J. Internet Services Appl.*, vol. 9, no. 1, pp. 1–28, Dec. 2018.

[36] S. Guo, F. Alamudun, and T. Hammond, "RésuMatcher: A personalized résumé-job matching system," *Expert Syst. Appl.*, vol. 60, pp. 169–182, Oct. 2016.

[37] B. Mobasher, "Data mining for web personalization," in *The Adaptive Web*. Cham, Switzerland: Springer, 2007, pp. 90–135.

[38] A. Alhozaimi and M. Almishari, "Arabic Twitter profiling for arabic-speaking users," in *Proc. 21st Saudi Comput. Soc. Nat. Comput. Conf. (NCC)*, Apr. 2018, pp. 1–6.

[39] S. Ostendorf, S. M. Müller, and M. Brand, "Neglecting long-term risks: Self-disclosure on social media and its relation to individual decision-making tendencies and problematic social-networks-use," *Frontiers Psychol.*, vol. 11, p. 2913, Oct. 2020.

[40] J. Tang, L. Yao, D. Zhang, and J. Zhang, "A combination approach to web user profiling," *ACM Trans. Knowl. Discovery Data*, vol. 5, no. 1, pp. 1–44, Dec. 2010.

[41] M. He, X. Wu, J. Zhang, and R. Dong, "UP-TreeRec: Building dynamic user profiles tree for news recommendation," *China Commun.*, vol. 16, no. 4, pp. 219–233, Apr. 2019.

[42] D. Zhu, Y. Wang, C. You, J. Qiu, N. Cao, C. Gong, G. Yang, and H. Min Zhou, "MMLUP: Multi-source & multi-task learning for user profiles in social network," *Comput., Mater. Continua*, vol. 61, no. 3, pp. 1105–1115, 2019.

[43] F. Zarrinkalam, M. Kahani, and E. Bagheri, "Mining user interests over active topics on social networks," *Inf. Process. Manage.*, vol. 54, no. 2, pp. 339–357, Mar. 2018.

[44] L. J. P. Reyes, N. B. Oviedo, E. C. Camacho, and J. M. Calderon, "Adaptable recommendation system for outfit selection with deep learning approach," *IFAC-PapersOnLine*, vol. 54, no. 13, pp. 605–610, 2021.

[45] M. Grzenda, S. Kaźmierczak, M. Luckner, G. Borowik, and J. Mańdziuk, "Evaluation of machine learning methods for impostor detection in web applications," *Expert Syst. Appl.*, vol. 231, Nov. 2023, Art. no. 120736.

[46] M. Hosseinzadeh, A. Hemmati, and A. M. Rahmani, "Clustering for smart cities in the Internet of Things: A review," *Cluster Comput.*, vol. 25, no. 6, pp. 4097–4127, Dec. 2022.

[47] S. M. Miraftabzadeh, C. G. Colombo, M. Longo, and F. Foiadelli, "K-means and alternative clustering methods in modern power systems," *IEEE Access*, vol. 11, pp. 119596–119633, 2023.

[48] A. M. Ikotun, A. E. Ezugwu, L. Abualigah, B. Abuhaija, and J. Heming, "K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data," *Inf. Sci.*, vol. 622, pp. 178–210, Apr. 2023.

[49] S.-H. Liao, R. Widowati, and P. Puttong, "Data mining analytics investigate Facebook live stream users' behaviors and business models: The evidence from Thailand," *Entertainment Comput.*, vol. 41, Mar. 2022, Art. no. 100478.

[50] Y. Qian, J. Shao, Z. Zhang, H. Leng, M. Ma, and Z. Li, "BERT-CK: A study of user profile classification based on BERT and CK-means+ fusion," *J. Intell. Fuzzy Syst.*, vol. 45, no. 3, pp. 4585–4597, Aug. 2023.

[51] A. Yassine, L. Mohamed, and M. A. Achhab, "Intelligent recommender system based on unsupervised machine learning and demographic attributes," *Simul. Model. Pract. Theory*, vol. 107, Feb. 2021, Art. no. 102198.

[52] A. S. Harish and C. Malathy, "Customer segment prediction on retail transactional data using K-means and Markov model," *Intell. Autom. Soft Comput.*, vol. 36, no. 1, pp. 589–600, 2023.

[53] B. Wang, J. Yin, Q. Hua, Z. Wu, and J. Cao, "Parallelizing K-means-based clustering on spark," in *Proc. Int. Conf. Adv. Cloud Big Data (CBD)*, Aug. 2016, pp. 31–36.

[54] B. Sun and H. Chen, "A survey of K nearest neighbor algorithms for solving the class imbalanced problem," *Wireless Commun. Mobile Comput.*, vol. 2021, no. 1, pp. 1–12, Jan. 2021.

[55] M. Ludewig and D. Jannach, "Evaluation of session-based recommendation algorithms," *User Model. User-Adapted Interact.*, vol. 28, nos. 4–5, pp. 331–390, Dec. 2018.

[56] M. Ludewig, N. Mauro, S. Latifi, and D. Jannach, "Empirical analysis of session-based recommendation algorithms: A comparison of neural and non-neural approaches," *User Model. User-Adapted Interact.*, vol. 31, no. 1, pp. 149–181, Mar. 2021.

[57] C. Kumar and M. Kumar, "User session interaction-based recommendation system using various machine learning techniques," *Multimedia Tools Appl.*, vol. 82, no. 14, pp. 21279–21309, Jun. 2023.

[58] X. Zhou, X. Liang, J. Zhao, A. Zhiyuli, and H. Zhang, "An unsupervised user identification algorithm using network embedding and scalable nearest neighbour," *Cluster Comput.*, vol. 22, no. S4, pp. 8677–8687, Jul. 2019.

[59] L. Berkani, S. Belkacem, M. Ouafi, and A. Guessoum, "Recommendation of users in social networks: A semantic and social based classification approach," *Expert Syst.*, vol. 38, no. 2, Mar. 2021, Art. no. e12634.

[60] Z. Wang, Q. Yan, Z. Wang, and X. Hei, "A weighted Naïve Bayes for image classification based on adaptive genetic algorithm," in *Proc. Int. Conf. Natural Comput., Fuzzy Syst. Knowl. Discovery*. Cham, Switzerland: Springer, 2022, pp. 543–550.

[61] Y. Pratama, A. R. Tampubolon, L. D. Sianturi, R. D. Manalu, and D. F. Pangaribuan, "Implementation of sentiment analysis on Twitter using Naïve Bayes algorithm to know the people responses to debate of DKI Jakarta governor election," *J. Phys., Conf. Ser.*, vol. 1175, Mar. 2019, Art. no. 012102.

[62] N. Jing, Z. Wu, S. Lyu, and V. Sugumaran, "Information credibility evaluation in online professional social network using tree augmented Naïve Bayes classifier," *Electron. Commerce Res.*, vol. 21, no. 2, pp. 645–669, Jun. 2021.

[63] M. Afzaal, M. Usman, and A. Fong, "Tourism mobile app with aspect-based sentiment classification framework for tourist reviews," *IEEE Trans. Consum. Electron.*, vol. 65, no. 2, pp. 233–242, May 2019.

[64] I. Wickramasinghe and H. Kalutarage, "Naive Bayes: Applications, variations and vulnerabilities: A review of literature with code snippets for implementation," *Soft Comput.*, vol. 25, no. 3, pp. 2277–2293, Feb. 2021.

[65] H. S. Laxmisagar and M. C. Hanumantharaju, "FPGA implementation of breast cancer detection using SVM linear classifier," *Multimedia Tools Appl.*, vol. 82, no. 26, pp. 41105–41128, Nov. 2023.

[66] X. Sun, Z. Huang, X. Peng, Y. Chen, and Y. Liu, "Building a model-based personalised recommendation approach for tourist attractions from geotagged social media data," *Int. J. Digit. Earth*, vol. 12, no. 6, pp. 661–678, Jun. 2019.

[67] Z. Peng, Q. Hu, and J. Dang, "Multi-kernel SVM based depression recognition using social media data," *Int. J. Mach. Learn. Cybern.*, vol. 10, no. 1, pp. 43–57, Jan. 2019.

[68] N. Kadam and S. K. Sharma, "Social media fake profile detection using data mining technique," *J. Adv. Inf. Technol.*, vol. 13, no. 5, pp. 518–523, 2022.

[69] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, "Dive into deep learning," 2021, *arXiv:2106.11342*.

[70] H. Al-Rikabi and B. Renczes, "Floating-point quantization analysis of multi-layer perceptron artificial neural networks," *J. Signal Process. Syst.*, vol. 96, nos. 4–5, pp. 301–312, May 2024.

[71] A. M. Ali, F. A. Ghaleb, M. S. Mohammed, F. J. Alsolami, and A. I. Khan, "Web-informed-augmented fake news detection model using stacked layers of convolutional neural network and deep autoencoder," *Mathematics*, vol. 11, no. 9, p. 1992, Apr. 2023.

[72] C. Gao, X. He, D. Gan, X. Chen, F. Feng, Y. Li, T.-S. Chua, and D. Jin, "Neural multi-task recommendation from multi-behavior data," in *Proc. IEEE 35th Int. Conf. Data Eng. (ICDE)*, Apr. 2019, pp. 1554–1557.

[73] C. Chen, X. Meng, Z. Xu, and T. Lukasiewicz, "Location-aware personalized news recommendation with deep semantic analysis," *IEEE Access*, vol. 5, pp. 1624–1638, 2017.

[74] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[75] J. Kang, H. Choi, and H. Lee, "Deep recurrent convolutional networks for inferring user interests from social media," *J. Intell. Inf. Syst.*, vol. 52, no. 1, pp. 191–209, Feb. 2019.

[76] G. Cucurull, P. Rodríguez, V. O. Yazici, J. M. Gonfaus, F. X. Roca, and J. Gonzàlez, "Deep inference of personality traits by integrating image and word use in social networks," 2018, *arXiv:1802.06757*.

[77] S. C. Guntuku, W. Lin, J. Carpenter, W. K. Ng, L. H. Ungar, and D. Preoţiuc-Pietro, "Studying personality through the content of posted and liked images on Twitter," in *Proc. ACM Web Sci. Conf.*, Jun. 2017, pp. 223–227.

[78] P. Wanda, "RunMax: Fake profile classification using novel nonlinear activation in CNN," *Social Netw. Anal. Mining*, vol. 12, no. 1, p. 158, Dec. 2022.

[79] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, A. M. Umar, O. U. Linus, H. Arshad, A. A. Kazaure, U. Gana, and M. U. Kiru, "Comprehensive review of artificial neural network applications to pattern recognition," *IEEE Access*, vol. 7, pp. 158820–158846, 2019.

[80] R. Cui, G. Agrawal, and R. Ramnath, "Tweets can tell: Activity recognition using hybrid gated recurrent neural networks," *Social Netw. Anal. Mining*, vol. 10, no. 1, pp. 1–15, Dec. 2020.

[81] T. M. Phuong, T. C. Thanh, and N. X. Bach, "Neural session-aware recommendation," *IEEE Access*, vol. 7, pp. 86884–86896, 2019.

[82] L. May Petry, C. Leite Da Silva, A. Esuli, C. Renso, and V. Bogorny, "MARC: A robust method for multiple-aspect trajectory classification via space, time, and semantic embeddings," *Int. J. Geographical Inf. Sci.*, vol. 34, no. 7, pp. 1428–1450, Jul. 2020.

[83] A. Alshehri, N. Khan, A. Alowayr, and M. Yahya Alghamdi, "Cyber-attack detection framework using machine learning and user behavior analytics," *Comput. Syst. Sci. Eng.*, vol. 44, no. 2, pp. 1679–1689, 2023.

[84] R. Chalehchaleh, M. Salehi, R. Farahbakhsh, and N. Crespi, "BRaG: A hybrid multi-feature framework for fake news detection on social media," *Social Netw. Anal. Mining*, vol. 14, no. 1, p. 35, Jan. 2024.

[85] M. Niepert, M. Ahmed, and K. Kutzkov, "Learning convolutional neural networks for graphs," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2016, pp. 2014–2023.

[86] J. Wen, L. Wei, W. Zhou, J. Han, and T. Guo, "GCN-IA: User profile based on graph convolutional network with implicit association labels," in *Proc. 20th Int. Conf. Comput. Sci.*, Amsterdam, The Netherlands. Cham, Switzerland: Springer, 2020, pp. 355–364.

[87] L. Pasa, N. Navarin, and A. Sperduti, "Polynomial-based graph convolutional neural networks for graph classification," *Mach. Learn.*, vol. 111, no. 4, pp. 1205–1237, Apr. 2022.

[88] M. Diao, Z. Zhang, S. Su, S. Gao, and H. Cao, "UPON: User profile transferring across networks," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2020, pp. 265–274.

[89] X. Xu, X. Zhao, M. Wei, and Z. Li, "A comprehensive review of graph convolutional networks: Approaches and applications," *Electron. Res. Arch.*, vol. 31, no. 7, pp. 4185–4215, 2023.

[90] S. Souravlas, S. Anastasiadou, and S. Katsavounis, "A survey on the recent advances of deep community detection," *Appl. Sci.*, vol. 11, no. 16, p. 7179, Aug. 2021.

[91] R. Xu, Y. Che, X. Wang, J. Hu, and Y. Xie, "Stacked autoencoder-based community detection method via an ensemble clustering framework," *Inf. Sci.*, vol. 526, pp. 151–165, Jul. 2020.

[92] V. Bhatia and R. Rani, "A distributed overlapping community detection model for large graphs using autoencoder," *Future Gener. Comput. Syst.*, vol. 94, pp. 16–26, May 2019.

[93] Y. Hao and F. Zhang, "An unsupervised detection method for shilling attacks based on deep learning and community detection," *Soft Comput.*, vol. 25, no. 1, pp. 477–494, Jan. 2021.

[94] P. Wang, Y. Fu, H. Xiong, and X. Li, "Adversarial substructured representation learning for mobile user profiling," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 130–138.

[95] K. Li, G. Lu, G. Luo, and Z. Cai, "Seed-free graph de-anonymiztiation with adversarial learning," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2020, pp. 745–754.

[96] S. Pan, R. Hu, S.-F. Fung, G. Long, J. Jiang, and C. Zhang, "Learning graph embedding with adversarial training methods," *IEEE Trans. Cybern.*, vol. 50, no. 6, pp. 2475–2487, Jun. 2020.

[97] D.-V. Vo, T.-T. Tran, K. Shirai, and V.-N. Huynh, "Deep generative networks coupled with evidential reasoning for dynamic user preferences using short texts," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 7, pp. 6811–6826, Jul. 2022.

[98] J. Cheng, Y. Yang, X. Tang, N. Xiong, Y. Zhang, and F. Lei, "Generative adversarial networks: A literature review," *KSII Trans. Internet Inf. Syst.*, vol. 14, no. 12, pp. 4625–4647, 2020.

[99] B. B. Sinha and R. Dhanalakshmi, "Evolution of recommender paradigm optimization over time," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 4, pp. 1047–1059, Apr. 2022.

[100] O. Choi and S. Y. Han, "Personalization of rule-based web services," *Sensors*, vol. 8, no. 4, pp. 2424–2435, Apr. 2008.

[101] H. Magadum, H. K. Azad, and H. Patel, "Music recommendation using dynamic feedback and content-based filtering," *Multimedia Tools Appl.*, pp. 1–20, Feb. 2024. [Online]. Available: https://link.springer.com/article/10.1007/s11042-024-18636-8#citeas

[102] D. Fernández, V. Formoso, F. Cacheda, and V. Carneiro, "A content-based approach to profile expansion," *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst.*, vol. 28, no. 6, pp. 981–1002, Dec. 2020.

[103] S. Reddy, S. Nalluri, S. Kunisetti, S. Ashok, and B. Venkatesh, "Content-based movie recommendation system using genre correlation," in *Proc. Smart Intell. Comput. Appl. Proc. 2nd Int. Conf. (SCI)*. Cham, Switzerland: Springer, 2019, pp. 391–397.

[104] P. Tatit, K. Adhinugraha, and D. Taniar, "Navigating the maps: Euclidean vs. road network distances in spatial queries," *Algorithms*, vol. 17, no. 1, p. 29, Jan. 2024.

[105] J. Shu, X. Shen, H. Liu, B. Yi, and Z. Zhang, "A content-based recommendation algorithm for learning resources," *Multimedia Syst.*, vol. 24, no. 2, pp. 163–173, Mar. 2018.

[106] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egyptian Informat. J.*, vol. 16, no. 3, pp. 261–273, Nov. 2015.

[107] C. M. Wu, D. Garg, and U. Bhandary, "Movie recommendation system using collaborative filtering," in *Proc. IEEE 9th Int. Conf. Softw. Eng. Service Sci. (ICSESS)*, Nov. 2018, pp. 11–15.

[108] M. Nilashi, O. Ibrahim, and K. Bagherifard, "A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques," *Expert Syst. Appl.*, vol. 92, pp. 507–520, Feb. 2018.

[109] Y. Su, X. Li, W. Tang, J. Xiang, and Y. He, "Next check-in location prediction via footprints and friendship on location-based social networks," in *Proc. 19th IEEE Int. Conf. Mobile Data Manage. (MDM)*, Jun. 2018, pp. 251–256.

[110] Z.-F. Peng, H.-R. Zhang, and F. Min, "IUG-CF: Neural collaborative filtering with ideal user group labels," *Expert Syst. Appl.*, vol. 238, Mar. 2024, Art. no. 121887.

[111] M. Jian, C. Zhang, T. Wang, and L. Wu, "Non-pairwise collaborative filtering," *Neural Process. Lett.*, vol. 55, no. 6, pp. 7627–7648, Dec. 2023.

[112] A. Torkashvand, S. M. Jameii, and A. Reza, "Deep learning-based collaborative filtering recommender systems: A comprehensive and systematic review," *Neural Comput. Appl.*, vol. 35, no. 35, pp. 24783–24827, Dec. 2023.

[113] M. F. Aljunid and M. Doddaghatta Huchaiah, "Multi-model deep learning approach for collaborative filtering recommendation system," *CAAI Trans. Intell. Technol.*, vol. 5, no. 4, pp. 268–275, Dec. 2020.

[114] L. Berkani and N. Boudjenah, "S-SNHF: Sentiment based social neural hybrid filtering," *Int. J. Gen. Syst.*, vol. 52, no. 3, pp. 297–325, Apr. 2023.

[115] K. N. Jain, V. Kumar, P. Kumar, and T. Choudhury, "Movie recommendation system: Hybrid information filtering system," in *Proc. 2nd Int. Conf. Intell. Comput. Inf. Commun. (ICICC)*. Cham, Switzerland: Springer, 2018, pp. 677–686.

[116] H. Wang, F. Zhang, J. Wang, M. Zhao, W. Li, X. Xie, and M. Guo, "RippleNet: Propagating user preferences on the knowledge graph for recommender systems," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2018, pp. 417–426.

[117] Y. Zhang, Q. Ai, X. Chen, and P. Wang, "Learning over knowledge-base embeddings for recommendation," 2018, *arXiv:1803.06540*.

[118] Q. Guo, F. Zhuang, C. Qin, H. Zhu, X. Xie, H. Xiong, and Q. He, "A survey on knowledge graph-based recommender systems," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 8, pp. 3549–3568, Aug. 2022.

[119] M. Zhu, D.-S. Zhen, R. Tao, Y.-Q. Shi, X.-Y. Feng, and Q. Wang, "Top-N collaborative filtering recommendation algorithm based on knowledge graph embedding," in *Proc. Int. Conf. Knowl. Manag. Organizations*, Zamora, Spain. Cham, Switzerland: Springer, 2019, pp. 122–134.

[120] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, "PathSim: Meta path-based top-K similarity search in heterogeneous information networks," *Proc. VLDB Endowment*, vol. 4, no. 11, pp. 992–1003, Aug. 2011.

[121] X. Yu, X. Ren, Y. Sun, Q. Gu, B. Sturt, U. Khandelwal, B. Norick, and J. Han, "Personalized entity recommendation: A heterogeneous information network approach," in *Proc. 7th ACM Int. Conf. Web Search Data Mining*, Feb. 2014, pp. 283–292.

[122] H. Zhao, Q. Yao, J. Li, Y. Song, and D. L. Lee, "Meta-graph based recommendation fusion over heterogeneous information networks," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2017, pp. 635–644.

[123] Z. Sun, J. Yang, J. Zhang, A. Bozzon, L.-K. Huang, and C. Xu, "Recurrent knowledge graph embedding for effective recommendation," in *Proc. 12th ACM Conf. Recommender Syst.*, Sep. 2018, pp. 297–305.

[124] X. Wen, S. Zhao, H. Wang, R. Wu, M. Qu, T. Hu, G. Chen, J. Tao, and C. Fan, "Multi-source multi-label learning for user profiling in online games," *IEEE Trans. Multimedia*, vol. 25, pp. 4135–4147, 2022.

[125] C. P. Santos, K. Hutchinson, V.-J. Khan, and P. Markopoulos, "Profiling personality traits with games," *ACM Trans. Interact. Intell. Syst.*, vol. 9, nos. 2–3, pp. 1–30, Sep. 2019.

[126] L. De Simone, D. Gadia, D. Maggiorini, and L. A. Ripamonti, "Design of a recommender system for video games based on in-game player profiling and activities," in *Proc. 14th Biannual Conf. Italian SIGCHI Chapter*, Jul. 2021, pp. 1–8.

[127] S. Liu, J. Xie, C. Zou, and Z. Chen, "User conditional hashtag recommendation for micro-videos," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2020, pp. 1–6.

[128] O. Lazzez, A. M. Qahtani, A. Alsufyani, O. Almutiry, H. Dhahri, V. Piuri, and A. M. Alimi, "DeepVisInterests: Deep data analysis for topics of interest prediction," *Multimedia Tools Appl.*, vol. 82, no. 26, pp. 40913–40936, Nov. 2023.

[129] S. Bennani, A. Maalel, H. Ben Ghezala, and A. Daouahi, "Integrating machine learning into learner profiling for adaptive and gamified learning system," in *Proc. Int. Conf. Comput. Collective Intell.* Cham, Switzerland: Springer, 2022, pp. 65–71.

[130] S. R. Rimitha, V. Abburu, A. Kiranmai, and K. Chandrasekaran, "Ontologies to model user profiles in personalized job recommendation," in *Proc. IEEE Distrib. Comput., VLSI, Electr. Circuits Robot. (DISCOVER)*, Aug. 2018, pp. 98–103.

[131] A. Rivas, P. Chamoso, A. González-Briones, R. Casado-Vara, and J. M. Corchado, "Hybrid job offer recommender system in a social network," *Expert Syst.*, vol. 36, no. 4, Aug. 2019, Art. no. e12416.

[132] P. Kaywan, K. Ahmed, Y. Miao, A. Ibaida, and B. Gu, "DEPRA: An early depression detection analysis chatbot," in *Proc. 10th Int. Conf. Health Inf. Sci. (HIS)*. Cham, Switzerland: Springer, 2021, pp. 193–204.

[133] B. Liu, Z. Xu, C. Sun, B. Wang, X. Wang, D. F. Wong, and M. Zhang, "Content-oriented user modeling for personalized response ranking in chatbots," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 26, no. 1, pp. 122–133, Jan. 2018.

[134] H. Qian, Z. Dou, Y. Zhu, Y. Ma, and J.-R. Wen, "Learning implicit user profile for personalized retrieval-based chatbot," in *Proc. 30th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2021, pp. 1467–1477.

[135] Y. Tai, Z. Sun, and Z. Yao, "Content-based recommendation using machine learning," in *Proc. IEEE 31st Int. Workshop Mach. Learn. Signal Process. (MLSP)*, Oct. 2021, pp. 1–4.

[136] W. Zeng, Y. Du, D. Zhang, Z. Ye, and Z. Dou, "TUP-RS: Temporal user profile based recommender system," in *Proc. 17th Int. Conf. Artif. Intell. Soft Comput.*, Zakopane, Poland. Cham, Switzerland: Springer, 2018, pp. 463–474.

[137] T. Bai, L. Zou, W. X. Zhao, P. Du, W. Liu, J.-Y. Nie, and J.-R. Wen, "CTRec: A long-short demands evolution model for continuous-time recommendation," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2019, pp. 675–684.

[138] X. Zhou, Z. Wang, X. Liu, Y. Liu, and G. Sun, "An improved context-aware weighted matrix factorization algorithm for point of interest recommendation in LBSN," *Inf. Syst.*, vol. 122, May 2024, Art. no. 102366.
[139] V. E. Carusotto, G. Pilato, F. Persia, and M. Ge, "User profiling for tourist trip recommendations using social sensing," in *Proc. IEEE Int. Symp. Multimedia (ISM)*, Nov. 2021, pp. 182–185.
[140] X. Chen, J. Chen, X. Lian, and W. Mai, "Resolving data sparsity via aggregating graph-based user–app–location association for location recommendations," *Appl. Sci.*, vol. 12, no. 14, p. 6882, Jul. 2022.
[141] A. Chakraborty and K. P. Gummadi, "Fairness in algorithmic decision making," in *Proc. 7th ACM IKDD CoDS 25th COMAD*, 2020, pp. 367–368.
[142] J. Sánchez-Monedero, L. Dencik, and L. Edwards, "What does it mean to 'solve' the problem of discrimination in hiring? Social, technical and legal perspectives from the U.K. on automated hiring systems," in *Proc. Conf. Fairness, Accountability, Transparency*, Jan. 2020, pp. 458–468.
[143] R. Moradi and H. Hamidi, "A new mechanism for detecting shilling attacks in recommender systems based on social network analysis and Gaussian rough neural network with emotional learning," *Int. J. Eng.*, vol. 36, no. 2, pp. 321–334, 2023.
[144] M. Nycyk, "From data serfdom to data ownership: An alternative futures view of personal data as property rights," *J. Futures Stud.*, vol. 24, no. 4, pp. 25–34, 2020.
[145] M. Hojati, C. Farmer, R. Feick, and C. Robertson, "Decentralized geoprivacy: Leveraging social trust on the distributed web," *Int. J. Geographical Inf. Sci.*, vol. 35, no. 12, pp. 2540–2566, Dec. 2021.
[146] A. K. Shrestha, J. Vassileva, and R. Deters, "A blockchain platform for user data sharing ensuring user control and incentives," *Frontiers Blockchain*, vol. 3, Oct. 2020, Art. no. 497985.
[147] S. Fu and Z. Yao, "Privacy risk estimation of online social networks," in *Proc. Int. Conf. Netw. Netw. Appl. (NaNA)*, Dec. 2022, pp. 1–8.
[148] T. Guo, K. Dong, L. Wang, M. Yang, and J. Luo, "Privacy preserving profile matching for social networks," in *Proc. 6th Int. Conf. Adv. Cloud Big Data (CBD)*, Aug. 2018, pp. 263–268.
[149] X. Yao, R. Zhang, and Y. Zhang, "Differential privacy-preserving user linkage across online social networks," in *Proc. IEEE/ACM 29th Int. Symp. Quality Service (IWQOS)*, Jun. 2021, pp. 1–10.
[150] S. J. De and A. Imine, "Privacy scoring of social network user profiles through risk analysis," in *Proc. Int. Conf. Risks Secur. Internet Syst.*, Dinard, France. Cham, Switzerland: Springer, 2018, pp. 227–243.
[151] I. Jekabsone, "Selected legal issues in online adult education: Compliance of online learning and teaching process with GDPR," *TalTech J. Eur. Stud.*, vol. 13, no. 2, pp. 46–62, Dec. 2023.
[152] F. S. Alrayes, A. I. Abdelmoty, W. B. El-Geresy, and G. Theodorakopoulos, "Modelling perceived risks to personal privacy from location disclosure on online social networks," *Int. J. Geographical Inf. Sci.*, vol. 34, no. 1, pp. 150–176, Jan. 2020.
[153] O. Amaral, M. I. Azeem, S. Abualhaija, and L. C. Briand, "NLP-based automated compliance checking of data processing agreements against GDPR," *IEEE Trans. Softw. Eng.*, vol. 49, no. 9, pp. 4282–4303, Jun. 2023.
[154] C. Novelli, M. Taddeo, and L. Floridi, "Accountability in artificial intelligence: What it is and how it works," *AI Soc.*, vol. 39, no. 4, pp. 1871–1882, Aug. 2024.
[155] J. Cobbe, M. S. A. Lee, and J. Singh, "Reviewable automated decision-making: A framework for accountable algorithmic systems," in *Proc. ACM Conf. Fairness, Accountability, Transparency*, Mar. 2021, pp. 598–609.
[156] T. D. Breaux and T. Norton, "Legal accountability as software quality: A U.S. data processing perspective," in *Proc. IEEE 30th Int. Requirement Eng. Conf. (RE)*, Aug. 2022, pp. 101–113.
[157] K. Biswas, P. Shivakumara, U. Pal, and T. Lu, "A new ontology-based multimodal classification system for social media images of personality traits," *Signal, Image Video Process.*, vol. 17, no. 2, pp. 543–551, Mar. 2023.
[158] P. M. Alamdari, N. J. Navimipour, M. Hosseinzadeh, A. A. Safaei, and A. Darwesh, "An image-based product recommendation for e-commerce applications using convolutional neural networks," *Acta Inf. Pragensia*, vol. 11, no. 1, pp. 15–35, 2022.
[159] S. Strukova, R. G. Marco, F. G. Mármol, and J. A. Ruipérez-Valiente, "Identifying professional photographers through image quality and aesthetics in Flickr," *Expert Syst.*, vol. 41, no. 4, Apr. 2024, Art. no. e13526.
[160] J. Chen, P. Ying, X. Fu, X. Luo, H. Guan, and K. Wei, "Automatic tagging by leveraging visual and annotated features in social media," *IEEE Trans. Multimedia*, vol. 24, pp. 2218–2229, 2022.
[161] Q. Cheng, Z. Ren, Y. Lin, P. Ren, Z. Chen, X. Liu, and M. D. de Rijke, "Long short-term session search: Joint personalized reranking and next query prediction," in *Proc. Web Conf.*, Apr. 2021, pp. 239–248.
[162] M. Wang, W. Chen, J. Xu, P. Zhao, and L. Zhao, "User profile linkage across multiple social platforms," in *Proc. 21st Int. Conf. Web Inf. Syst. Eng. (WISE)*, Amsterdam, The Netherlands. Cham, Switzerland: Springer, Oct. 2020, pp. 125–140.
[163] B. Treves, M. R. Masud, and M. Faloutsos, "RURLMAN: Matching forum users across platforms using their posted URLs," in *Proc. Int. Conf. Adv. Social Netw. Anal. Mining*, Nov. 2023, pp. 484–491.
[164] X. Zhou and J. Yang, "Matching user accounts based on location verification across social networks," *Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería*, vol. 36, no. 1, pp. 1–7, 2020.
[165] Y. Xiao, Y. Jia, X. Cheng, S. Wang, J. Mao, and Z. Liang, "I know your social network accounts: A novel attack architecture for device-identity association," *IEEE Trans. Dependable Secure Comput.*, vol. 20, no. 2, pp. 1017–1030, Mar. 2023.
[166] N. Wiedemann, K. Janowicz, M. Raubal, and O. Kounadi, "Where you go is who you are: A study on machine learning based semantic privacy attacks," *J. Big Data*, vol. 11, no. 1, pp. 1–31, Mar. 2024.
[167] A. Brandão, R. Mendes, and J. P. Vilela, "Prediction of mobile app privacy preferences with user profiles via federated learning," in *Proc. 12th ACM Conf. Data Appl. Secur. Privacy*, Apr. 2022, pp. 89–100.
[168] Z. Huo, T. Wang, Y. Fan, and P. He, "Privacy-preserving global user profile construction through federated learning," *Int. J. Comput. Sci. Eng.*, vol. 26, no. 2, p. 199, 2023.

WENBO WU is currently pursuing the Ph.D. degree with the Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru, Malaysia. He has over ten years of experience in software development. His research interests include machine learning, big data, social network analysis, and human–computer interaction.

MASITAH GHAZALI received the Ph.D. degree in computer science (human–computer interaction) from Lancaster University, U.K., in 2002. She is currently working as an Associate Professor with Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia, Kuala Lumpur. With a focus on persuasive technology, physicality, and mobile-BCI, she tackles challenges in sustainable behavior, digital-physical interaction, and emotion detection and expression. Her expertise also extends to UI/UX Projects, both within and outside the university. She has published for more than 110 articles with 50 being indexed publications.

SHARIN HAZLIN HUSPI received the Ph.D. degree in computer science from RMIT University, Australia. She is currently working as a Senior Lecturer with the Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru, Malaysia. She has more than 20 articles relevant to her research interests. Her research interests include natural language processing (NLP), data analytics, text mining and analytics, and user centered evaluation.