IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 29 September 2022, accepted 13 October 2022, date of publication 19 October 2022, date of current version 26 October 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3215732

RESEARCH ARTICLE

User Demographic Prediction Based on the Fusion of Mobile and Survey Data

XINGYU CHEN, YE GUO, HONGLEI XU, HONGYAN YAN, AND LIN LIN
Department of User and Market Research, China Mobile Research Institute, Beijing 100032, China
Corresponding author: Ye Guo (guoye36@outlook.com)

ABSTRACT The user demographic prediction problem is one of the critical processes in the construction of user profiles, which is of great significance for understanding users' characteristics and attributes. Most of the prior works on this problem either used only single-source data or employed a hard-matching method to handle multi-source data. These methods will result in a great loss of data and information in many circumstances, which may affect the model's accuracy as well as the application scenarios. In order to solve these problems, this paper proposes a framework for user demographic prediction based on mobile and survey data, and presents a Deep Structured Fusion Model (DSFM) using neural networks with attention mechanisms to perform data fusion by comparing user similarity between two heterogeneous datasets. We examine the effectiveness of the framework and the fusion model on a real-world mobile dataset with almost one billion users, using a survey dataset containing 29,809 users' questionnaire results as an additional information source to predict users' age and gender. Our framework achieves excellent results on these datasets, increasing the prediction accuracy of gender and age by up to 3.23% and 5.21% compared to the best baseline model.

INDEX TERMS User demographic prediction, mobile big data, survey data, data fusion, deep learning.

I. INTRODUCTION

User demographics such as age, gender, and income are one of the essential components in constructing user profiles. The user profile is a model of user characteristics abstracted from information such as personal information, living habits, and consumption patterns. There are many real-world applications of it, some of which are shown in Fig. 1. One of the major applications is precision marketing for advertising. Different user groups can be identified early on through the study of user profiles, and later on, depending on the advertiser's demands, the target groups can be effectively addressed to achieve accurate advertising [1], [2], [3]. It is also feasible to use them as features to input click-through rate or conversion rate prediction models to improve their performance in advertising [4], [5]. Another application of user profiles is recommendation, where they can be used to evaluate different users' interest levels in various products in order to construct a personalized recommendation system [6], [7], [8], [9]. Furthermore, they can also be employed in decision-making research. It is possible to use them to pinpoint the target customer groups in market segments in finance, Internet, biomedicine, and other industries, thus clarifying product positioning and assisting decision-making [10], [11], [12]. All of these examples are down- stream applications for user profiles and can benefit from them.

One of the most important steps in constructing a user profile is determining his demographics. In general, user demographics can be obtained directly from the registration information. However, in practice, some of the required demographics are often complex and difficult to obtain directly. Moreover, owing to subjective and objective factors such as privacy protection or user refusal to provide, demographics that constitute the user profile are likely to be missing in some degree. And then, it is necessary to analyze and predict them based on the user's behavior data. As a result, research on user demographic prediction is of great significance for creating user profiles.

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

Many prior works on user demographic prediction have been conducted, but these works still have two significant limitations. First, most of these works only rely on single-source data to train the prediction model [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33]. Some of these works [20], [21], [28] also use multiple datasets, but only to test the performance of the prediction model in different situations, and the training process is still carried out on the single-source data. Nevertheless, data features from a single source are relatively constrained and cannot fully reflect user attributes. If demographic prediction can be performed using multi-source data to enrich the data volumes and features, the results will be more certain. Second, although some works do use heterogeneous data-sets from several sources at the same time to train the prediction model, the multi-source data used in these works are merged in a hard-matching method which does not achieve data fusion in the true sense [12], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45]. The hard-matching method refers to extracting users or features that overlap across multiple datasets, and then splicing them together to create a new dataset for model training. For those parts that do not overlap, they can only be discarded. Since the information sources of these datasets are different, the users and features in each dataset will be somewhat diverse in most practical circumstances. If we employ this hard-matching method to combine the multi-source data, a lot of information could be wasted, and it will be difficult to enrich the number of input users and features at the same time, which will be less helpful to improve the prediction model's performance.

To address these limitations, in this paper we propose a method for user demographic prediction based on mobile big data and survey data. Mobile big data are naturally generated by the interaction between people and heterogeneous mobile sources such as smartphones and sensors [46], including communication consumption, app usage, trajectory, etc. The number of users in mobile data could reach almost one billion, yet the data features are fixed and constrained. Survey data are proactively gathered from the user side through forms like questionnaires and interviews. Although the number of survey user is generally small, the data features are abundant since the survey questionnaire allows for flexible setting. As two heterogeneous datasets, the features in mobile and survey data have distinct ranges and depths. In consequence, we are able to achieve better results in user demographic prediction if these two datasets can be used simultaneously. Based on the mobile and survey data, our demographic prediction framework is as follows: firstly, design a questionnaire according to the predicted demographic; then carry out a sample survey on some users of the mobile dataset to get their demographics and other information; finally, extend the demographics of this survey participants to all mobile users to realize the demographic prediction for all users through the processing of models and algorithms. In addition, considering that mobile data and survey data are heterogeneous,

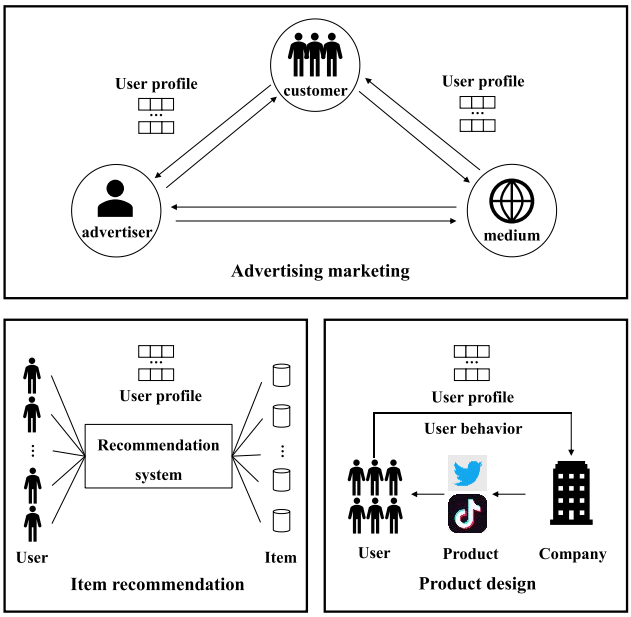


FIGURE 1. The applications of user profiles.

features in the datasets differ greatly, so it is impossible to directly merge these two datasets together. Therefore, this paper also proposes a Deep Structure Fusion Model (DSFM) using neural networks with attention mechanisms to achieve the fusion of two heterogeneous datasets by modeling the data fusion as a binary classification problem. On this basis, our framework employs DSFM to calculate the similarity scores between users of different datasets and then utilizes these scores as weights to determine users' demographics of the mobile dataset.

In comparison to previous works, our framework can predict demographics based on multiple heterogeneous datasets simultaneously, while the users and features from different datasets do not need to correspond one-to-one. Unlike the hard-matching method in prior works, our framework can compare the similarity of users in different datasets using DSFM, and accomplish the soft matching of users and features based on the similarity score provided by the model, thus realizing the real fusion between datasets. The advantage of this approach is that it allows the prediction model to make full use of all users and features in diverse datasets, avoiding discarding and wasting data information. This increases the number of input samples and enriches the range of features, which in turn leads to the improvements of model performance. Our framework has been tested using real-world mobile and survey data, and the experiments show that our approach outperforms all baselines in the user demographic prediction problem.

The main contributions of this paper are summarized as follows:

- We formulate the user demographic prediction as a data fusion problem, and achieve the fusion of mobile and survey data in a soft-matching method by comparing similarity scores between users from these two datasets.

TABLE 1. A summary of user demographic prediction studies.

References	Type of the prediction model	Single-source data or multi-source data	Concatenation methods of datasets
Dogan *et al.* [13], Bwambale *et al.* [14], Al-Ghadir *et al.* [15], Nguyen *et al.* [16], Malmi *et al.* [17], Matz *et al.* [18], Kaur *et al.* [19], Bouadjenek *et al.* [20], Jain *et al.* [21], Urbancokova *et al.* [22]	Machine learning	Single-source data	No concatenation
Dong *et al.* [34], Choi *et al.* [35], Jahani *et al.* [36]	Machine learning	Multi-source data	User concatenation
Kim *et al.* [12], Ulges *et al.* [37], Al-Zuabi *et al.* [38], Blumenstock *et al.* [39], Hirt *et al.* [40], Rosenfeld *et al.* [41], Culotta *et al.* [42], Tadesse *et al.* [43]	Machine learning	Multi-source data	Feature concatenation
Priadana *et al.* [23], Pandya *et al.* [24], Liu *et al.* [25], Van hamme *et al.* [26], Suman *et al.* [27], Antipov *et al.* [28], Kaushik *et al.* [29], Suh *et al.* [30], Qureshi *et al.* [31], Oh *et al.* [32], Guimaraes *et al.* [33]	Deep learning	Single-source data	No concatenation
Wood-Doughty *et al.* [44]	Deep learning	Multi-source data	User concatenation
Figueroa *et al.* [45]	Deep learning	Multi-source data	Feature concatenation

- We present a data fusion model named DSFM using neural networks with attention mechanisms, which realizes the soft matching of multi-source heterogeneous datasets by modeling the data fusion as a binary classification problem.
- We conduct experiments using real-world anonymous mobile and survey datasets to verify the effectiveness of our proposed framework and model.

The remaining sections of this paper are organized as follows. Section II discusses some related work on user demographic prediction and fusion of mobile and survey data. Section III defines the problem and presents a solution framework. Section IV describes the methods of preprocessing and modeling in detail. Section V discusses the experimental results, and Section VI summarizes the overall work of this paper at last.

II. RELATED WORK

User demographic prediction has always been a focus of research because it can reveal the user's attribute characteristics and behavior patterns, which are crucial for user and market research. Many prior works pay attention to the user demographic prediction problem. We have conducted a lot of research on related works, selected some representtative studies, and classified them according to their modeling methods and datasets, as shown in Table 1. According to the specific prediction model selection, we can categorize these works into two groups: those that predict user demographics using traditional machine learning models such as support vector machine, bayes network and decision trees [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43] or deep learning models such as CNN, GRU and Resnet [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [44], [45]. The choice of model is closely related to the particular dataset and application scenario.

Malmi et al. [17] made user demographic predictions based on the user's installed app information. They used a logistic regression model to predict demographic attributes including user age, gender, race, and income on a dataset of 3,760 users. Matz et al. [18] conducted user income prediction using a lasso regression model on a dataset of 2,623 users based on Twitter data. Liu et al. [25] proposed a DEREK framework based on neural networks for user demographic prediction using user rating data for movies. They adopted a heuristic data generation method in the model, and achieved better results than traditional machine learning models on the MovieLens Datasets which provides around 1,000,000 ratings for movies. Suman et al. [27] proposed a multimodal emotion detection model which achieved good results on the PAN2018 dataset using deep learning models such as Resnet and GRU to process 49,000 images and 490,000 text data respectively. Compared to traditional machine learning models, deep learning models often achieve better results when processing large volumes of data, especially unstructured data such as images, text, and id features [47].

As presented in the introduction section, most of the previous studies suffer from two main limitations. First, we can

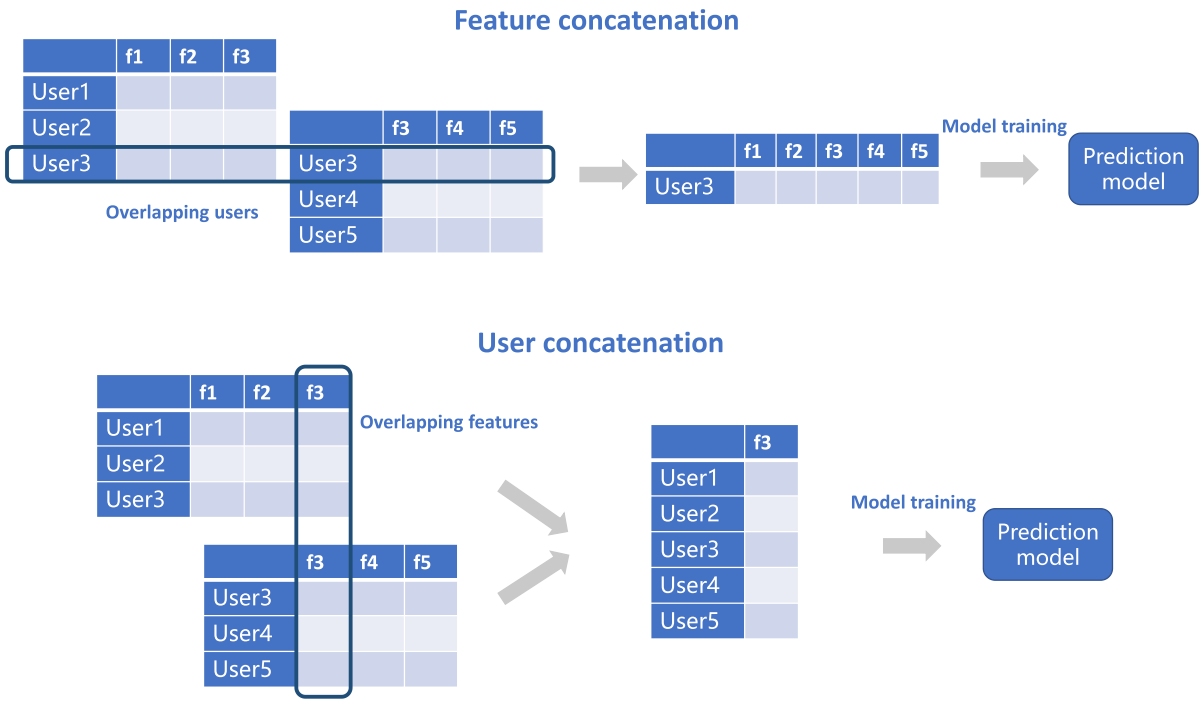


FIGURE 2. The flow of hard-matching method which can be divided into feature concatenation and user concatenation.

see from Table 1 that many studies only utilize single-source data for user demographic prediction [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], e.g. using only twitter [16], [18], [24] or mobile data [14], [17]. In this case, the performances of prediction models are highly limited by the richness of data features. For example, [23] used image data for user gender prediction and its model accuracy was only 70.11%, whereas [38] used data generated from cell phones for gender prediction and its model accuracy could reach 85.60%. It can be seen that the model performance is closely related to the dataset. Factors such as data type, volume and feature will affect the upper bound of the demographic prediction model [16].

Besides that, although some studies have used multi-source data to train prediction models, the methods of these studies to deal with multi-source data are all direct hard matching, that is, extracting users or features that overlap across datasets, and then combining them into a new dataset for model training. As shown in Table 1, according to the specific dataset concatenating methods for multi-source data, we can divide these studies into two categories, one using feature concatenation and the other using user concatenation. We show the exact flow of these two methods in Fig. 2. Feature concatenation is to filter out overlapping users in different datasets, and then aggregate the features of these users in multi-source datasets into one set. It is mainly used in the case that the user entities in different datasets are almost identical but have diverse features. As can be seen from Fig. 2, if the users in two datasets are too dissimilar, adopting this concatenation method increases the breadth of the model input features but results in the loss of a considerable quantity of user data. In contrast, user concatenation, which is shown in the lower part of Fig. 2, is to select the overlapping features in different datasets, and then aggregate the user entities in multi-source datasets under these features into one set. It is mainly used in the case where the user entities in multi-source datasets are different, but the features are almost identical. Fig. 2 shows that if the features of the two datasets are too dissimilar, using this method increases the number of model training samples while losing a significant amount of features. In general, the hard-matching methods used to deal with multi-source datasets in previous studies, whether it is feature concatenation or user concatenation, have poor application scope and effectiveness. In practice, due to the variations in the information sources and collection methods of multi-source datasets, the user entities and features between these datasets are quite different. If hard-matching methods are used, a large amount of data information will be wasted, which is detrimental to the improvement of prediction model performance.

Compared with the previous studies, the framework proposed in this paper which predicts user demographics based on the fusion of mobile and survey data is significantly more practical. Our framework uses the DSFM to compare the similarity of user entities from heterogeneous datasets, and then implements soft matching between multi-source datasets based on similarity scores. As shown in Fig. 3, this soft-matching method can make full use of user and feature information from disparate datasets with nearly no information loss, allowing for data fusion in the true sense and

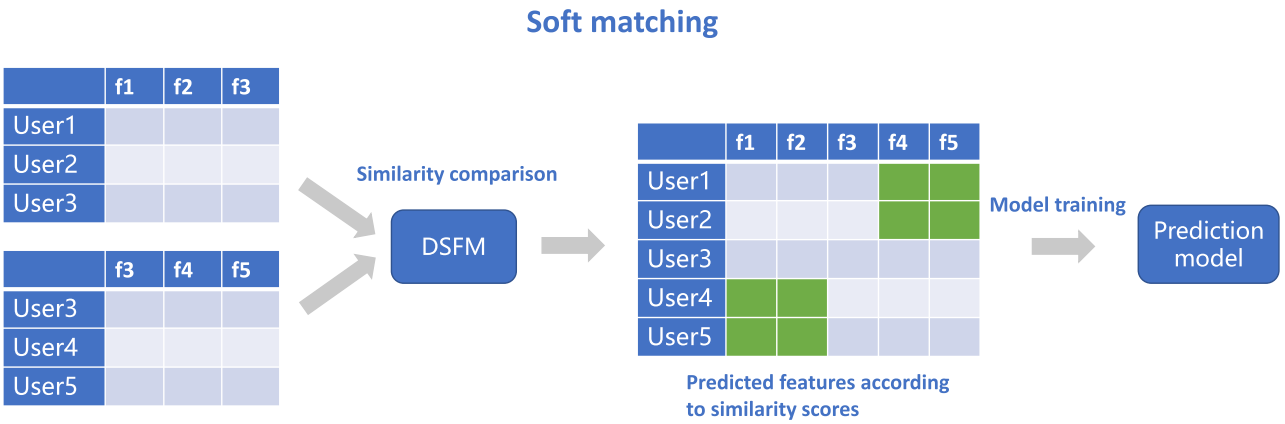


FIGURE 3. The flow of soft-matching method using the deep structure fusion model (DSFM).

sufficient data support for the prediction model. Section III and section IV will go over the exact implementation technique in depth.

In order to verify the performance of the model framework proposed in this paper, we will use mobile and survey data for experimental verification of user demo- graphic prediction. Although the era of big data has long arrived, user surveys via questionnaires still play an important role in user analysis, social research, and other fields. Many studies have demonstrated that big data cannot replace survey data fully. They can collaborate and play complementary roles. Therefore, if the mobile big data and survey data can be used for user demographic prediction at the same time, the results' credibility would be considerably increased. Some studies attempted to combine them together [39], [48], [49], [50], but these studies did not provide a model approach for fine-grained fusion of these two heterogeneous data. Reference [39] used mobile data and survey data from questionnaires to conduct user research on 856 mobile users, and built a model to predict users' wealth, income and other attributes successfully. Although the model has achieved good results, this study does not utilize a data fusion model, but merges the mobile and survey data of these 856 users directly and trains the model on the combined dataset of these users. This strategy is exactly the feature concatenation in the hard-matching method we introduced earlier. The disadvantage of this strategy is that the amount of data in the training set is greatly limited by the number of survey participants. If the number of survey participants is small, the model's performance will be severely hampered. If we apply the model to millions of people after just training it on a few hundred, it is likely to cause uncontrollable errors.

In this paper, we present a data fusion model to combine mobile and survey data. The model's output is the similarity score of a mobile user and a survey participant, from which we can deduce mobile users' demographics. More importantly, when we train the data fusion model, we also input non-survey users among mobile users into the model in addition to the survey participants. The model's input is in pair-wise form, and we can select one person from each of the two datasets to form a user pair. We can generate training samples that are hundreds of times larger than survey participants by adjusting the ratio of positive and negative samples, which will be discussed in detail in section IV. In this way, the model will outperform the model trained on survey participants only. Instead of just splicing the features of the mobile and survey datasets for the sample survey participants, our approach accomplishes a full fusion of these two datasets.

III. OVERVIEW

In this section, we formally define the problems and introduce the framework for their solution. In addition, since the following sections of this paper will use many symbols and terminologies, we list them with description in Table 2.

A. PROBLEM DEFINITION

The purpose of our framework is to predict user demographics based on the fusion of mobile and survey data. As a result, we define the data fusion problem first.

The Fusion of Mobile and Survey Data (FuMSD) problem focuses on correlating all users of mobile data with users of survey data. Since the users of the two datasets are not in one-to-one correspondence, we use the similarity score between users to achieve association and fusion. Next, we will further elaborate on the FuMSD problem. Given two multi-source heterogeneous datasets U and V, U contains features of p mobile users, denoted as $U\{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots \boldsymbol{u}_p\}$ where $\boldsymbol{u}_i = (f_1^{u_i}, f_2^{u_i}, \ldots)$ and f represents specific features, V contains features of q survey users, denoted as $V\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots \boldsymbol{v}_q\}$ where $\boldsymbol{v}_i = (f_1^{v_i}, f_2^{v_i}, \ldots)$. The survey users are a subset of mobile users, any user in V can be found to have one and only one user matching it in U, and these two users correspond to the same person in reality. In most cases, the number of survey users will be substantially lower than the number of mobile users, that is $p \gg q$. For mobile users who are not in V, we cannot directly find their

TABLE 2. The list of symbols and terminologies used in this paper.

Symbol	Description	Symbol	Description
U	Mobile dataset	U_{match}	The user set returned by the matching module
V	Survey dataset	e_i	An input vector of the layers in neural networks
p	Number of the users in the mobile dataset	out_{cat}	Output vector of a concatenation layer
q	Number of the users in the survey dataset	out_{pool}	Output vector of a pooling layer
u_i	The i-th user in the mobile dataset	$o(.)$	Outer product calculation
v_i	The i-th user in the survey dataset	$out_{ac-unit}$	Output vector of an activation unit
$f_j^{u_i}$	The j-th feature of user u_i in the mobile dataset	$\overline{s_i}$	Similarity scores divided by the sum
$f_j^{v_i}$	The j-th feature of user v_i in the survey dataset	s_i^{new}	Similarity corrected by sampling rate
T	Data fusion result of the mobile dataset and survey dataset	c	Sampling rate
s_i^j	Similarity score between the j-th mobile user and the i-th survey user	DSFM	Deep Structure Fusion Model
L	The user demographic	FuMSD	The Fusion of Mobile and Survey Data
r_i^L	The value under demographic L in the survey dataset	UDP	User Demographic Prediction
R_i^L	The value under demographic L in the mobile dataset	ARPU	Average Revenue Per User
$w_{i,j}$	Level of usage of the i-th app installed by user u_i	DOU	Discharge of usage
M_i	Total monthly data flow of the i-th app on all users	MOU	Minutes of usage
Z_i	Number of monthly live users of the i-th app	LR	Logistic Regression
$m_{i,j}$	Monthly data flow of user u_j on the i-th app	SVM	Support Vector Machine
Ω	The set of all apps installed by one user	LightGBM	Light Gradient Boosting Machine
h_i	Encoding of the i-th question	ANN	Artificial Neural Network
g_i	Number of options of the i-th question	RF	Random Forests
H	Encoding of the whole questionnaire	GBDT	Gradient Boosting Decision Tree

corresponding users. Instead, we utilize the similarity score between these users and survey users in V to achieve the data fusion. In this way, our goal changes to calculating similarity scores with all users in V for all users in U and getting $T = \{[u_{a_1}, (s_1^{a_1}, s_2^{a_1}, \ldots s_q^{a_1})], \ldots [u_{a_p}, (s_1^{a_p}, s_2^{a_p}, \ldots s_q^{a_p})]\}$, where $s_i^{a_j}$ represents the similarity score between the a_j-th mobile user and the i-th survey user, $i \in [1, q]$, $a_j \in [1, p]$. The next step is to process this set of similarity scores.

After solving the FuMSD problem, we can use the results of similarity score to solve the User Demographic Prediction (UDP) problem. The framework of this paper to address the UDP problem is to extend the survey participants' demographics to mobile users based on the fusion result T after solving the FuMSD problem. Suppose that we need to predict the value of all p mobile users under the demographic L, which is a categorical feature with k kinds of categories. We obtain the values of q survey participants under demogra-phic L that is $\{r_1^L, r_2^L, \ldots r_q^L\}$ where $1 \leq r_i^L \leq k$ through the survey questionnaire, and then fuse the mobile and survey data to get the result T. Based on the similarity scores in T, we set appropriate rules to determine the values under the demographic L of all p mobile users $\{R_1^L, R_2^L, \ldots R_p^L\}$ where $1 \leq R_i^L \leq k$.

B. SOLUTION FRAMEWORK

The solution framework proposed in this paper is shown in Fig. 4, and the overall process is divided into two stages. The first stage is to train a data fusion model for mobile and survey data. We use a matching module to filter a portion of users with high similarity scores to the survey participants through some strong correlation features, and put these users into the fusion model for training. The fusion model DSFM is a neural network model with attention mechanisms. The input of the model is in pair-wise form, and we select one person from

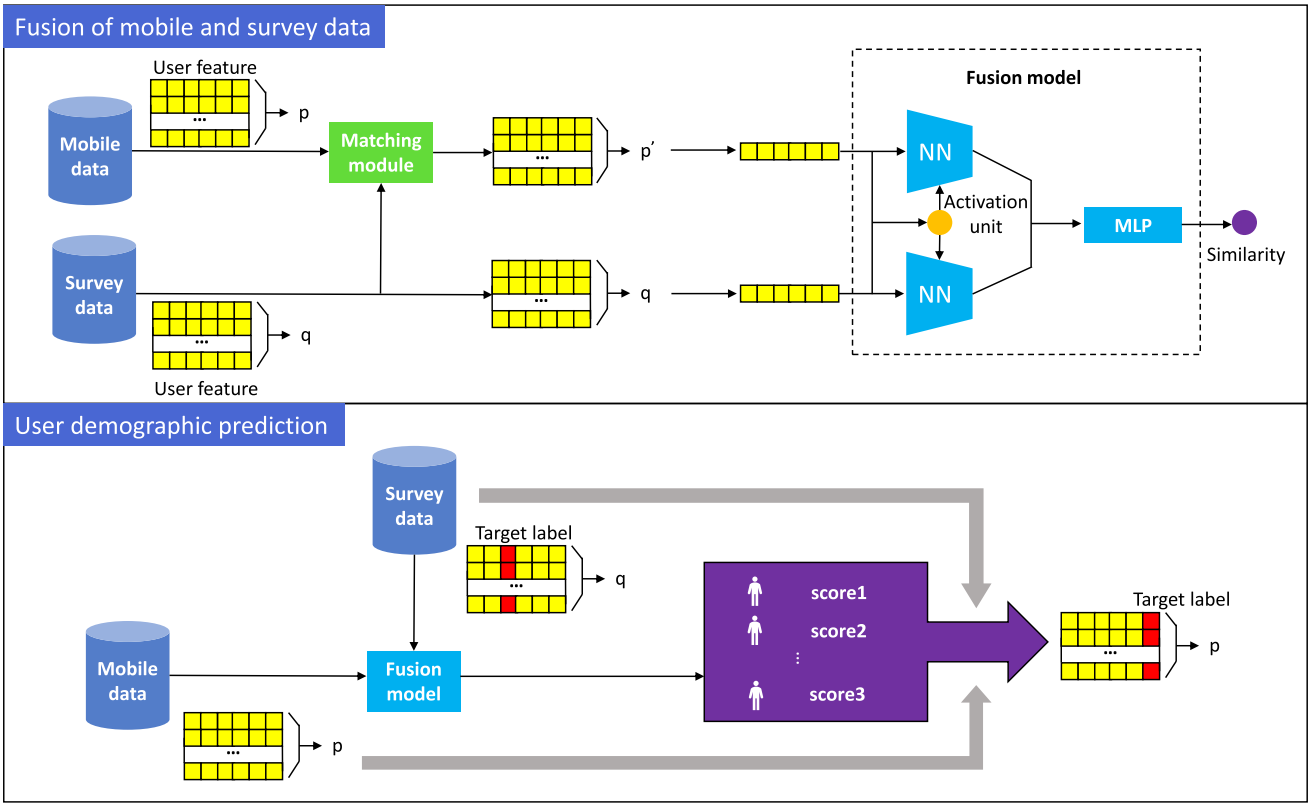


FIGURE 4. The framework of this paper. It mainly contains two stages and solve the UDP problem based on the fusion of mobile and survey data.

each dataset to assemble the user pair input into the model. The model's output is the similarity score of the two users in one input pair.

After the first stage of model training, the next stage of user demographic prediction in mobile dataset can be performed. For each mobile user, the trained data fusion model is utilized to determine the similarity score with all survey users. Based on this set of scores, the target demo- graphics in the survey data are mapped to the mobile data, and the value of each mobile user on the target attribute can be calculated to fulfill the purpose of demographic prediction.

IV. METHODOLOGY

A. DATASET DESCRIPTION

The purpose of this paper is to predict user demographics using mobile big data, and survey dataset is used as an additional source to support the prediction task via data fusion. We run experiments using a real-world mobile dataset and a survey dataset, both of which are encrypted and anonymous.

The mobile dataset used in this paper contains almost one billion user records collected in June 2022. Each piece of data in the collection comprises information about a specific person. The features of this dataset can be divided into three categories: personal information features, communication features, and mobile device features. The details are shown in Table 3. Personal information features refer to the user's own characteristics and attributes, mainly including the user's encrypted ID, age, gender, and city. Age and gender will be used as the demographics to be predicted in this paper. Communication features refer to the user's communication package information on mobile devices, mainly including communication expenditure, call duration, user star, and so on. Mobile device features refer to the physical hardware and software information of the mobile device such as device brand, camera pixel, and app usage.

The survey dataset is made up of the results from a questionnaire survey conducted on a sample of mobile users, comprising the questionnaire questions, associated options, and the response of 29,809 participants. These 29,809 participants also exist in the mobile datasets and can be linked by user ID. The collection time of this survey is also June 2022. There are 18 questions in the questionnaire, which can be divided into three categories according to their contents, namely personal information questions, communication questions, and app questions. We design these questions to explore user information in the corresponding fields, and these three categories of questions also correlate to the three feature categories in the mobile dataset. Considering the paper length and questionnaire's copyright, we only provide a few items from the survey in Table 4, and the corresponding options are in parentheses after the question. Some of these questions are identical to the user features in the mobile

TABLE 3. Features of the mobile dataset.

Personal information features	Communication features	Mobile device features
· User ID	· Average Revenue Per User (ARPU)	· List of used apps' name
· Age	· Discharge of usage (DOU)	· List of used apps' category
· Gender	· Minutes of usage (MOU)	· List of used apps' data flow
· City	· 5g user or not	· Device brand
	· Package brand	· 5g device or not
	· Package data flow	· Camera pixels
	· Package minutes	
	· Package cost	
	· User star	
	· In network duration	
	· Bandwidth	
	· Number of secondary cards	

TABLE 4. Questions and options of the survey questionnaire.

Personal information questions	Communication questions	App questions
· What is your gender? (male or female)	· When was the last time you applied for a phone card? (undergraduate, postgraduate, after graduation, …)	· What types of apps do you currently have membership in? (video apps, music apps, social apps, shopping apps, …)
· Which city do you live in?	· What is the reason for your last phone card change? (change a new phone， change a city, promotions, …)	· What types of benefits do you hope to obtain in the future? (sports, communication, entertainment, …)
· How old are you? (16-25, 26-35,…)		

dataset, such as gender, city, age, device brand and so on. However, most of the remaining questions cannot be directly related to the features but have a certain correlation. From one point of view, we can consider the questions, corresponding options, and user answers as the features of survey data.

According to the degree of correlation between the features of mobile and survey data, we can divide these features into directly related features and non-directly related features. Taking Table 3 and Table 4 as an example, the directly related features include age, gender, package brand and device brand. These features are directly queried in the questionnaire and can be easily associated with the corresponding features in the mobile dataset. As for non-directly related features, for example, there is a question "What types of apps do you currently have membership in?" in the questionnaire, this question asks about the user's membership status, but there are no features about app membership in the mobile dataset, only app flow information, so it cannot be directly related. However, from another perspective, we may learn which type of app one person is willing to paying for in order to have a better experience based on his response to this question. This person is likely to use this type of apps more frequently and the data flow spent on it will also be relatively more than the others, indicating that these two features are not completely independent, but there is a certain correlation.

B. DATA PREPROCESSING

In this paper, we predict user demographics based on the mobile and survey data. However, the mobile dataset mainly contains structured data, whereas the survey dataset is text data. In order for the model to process and analyze data more conveniently, we need to perform a series of preprocessing operations on these two original datasets respectively.

1) PREPROCESSING OF THE MOBILE DATASET

According to the introduction in the previous part, features in the mobile dataset can be divided into personal information features, communication features, and mobile device features from the perspective of their content. However, in the preprocessing stage, we need to classify them into categorical features, numeric features, and ID features according to the difference in their values. Categorical features are those that have discrete and limited values and can be separated into ordinal features and disorderly features according to whether the feature values have a size order. Ordinal features include user star, number of secondary cards, and so on, while disorderly features include device brand, package brand, etc. Numerical features are those that have a continuous value distribution and have intrinsic order, and the range of values can be the entire real number space with practical meaning,

such as MOU and DOU, which can be any value larger than or equal to 0. Compared with categorical features, ID features have more sparse values. The ID features in the mobile dataset include a list of used app names and categories. The method we preprocess these various types of features will differ as well.

The data preprocessing stage in this paper mainly includes two steps, which are outlier processing and feature encoding.

When dealing with outliers, for categorical features, outliers mainly refer to null values generated during data collection and transmission process. Our processing strategy for this data is to use the category with the most occurrences under the feature to fill. For numerical features, outliers include not only null values but also data with values that are either too large or too little. We use boxplots to filter out data with abnormal values, and use the average to fill in the null values. Since the ID features are presented as a list, there is no outlier in the general sense.

In the encoding step, we employ label encoding to handle ordinal categorical features. For features with k categories, each category will be assigned a value between 0 and k-1. The unordered categorical features are encoded using one-hot encoding. At this time, features with k categories will be converted into k binary features, with just one of these k features being 1 and the others being 0. For numerical features, the model can typically handle them directly, but in many circumstances, if the numerical features are not encoded and input directly, the model will be unable to learn the information of entire numerical domain due to their weak representation ability [51], so encoding processing is also necessary for numerical features. The encoding strategy for numerical features is discretization. There are many discretization methods, the most common of which are equal distance discretization, logarithm discretization [52], and entropy-based discretization [53], [54]. In this paper, equal distance discretization is used for features with a relatively uniform value distribution, such as in-network duration, while logarithm discretization is used for features whose value distribution is close to the normal distribution type, such as MOU and DOU. After discretization, numerical features become ordinal categorical features, which can be encoded using label encoding. As for ID features, including a list of apps' names and categories, we employ tokenization to handle with them, which refers to the processing method of text features. After tokenization, the app names and categories of text type will be turned into numerical type.

During the tokenization process, we will remove apps that are used too frequently or too infrequently, which is detrimental to the model's learning of user attributes. Apps that are used too frequently are essentially some commonly used software that cannot reflect the unique personal attributes of users; while apps that are used too infrequently have too few occurrences, it is difficult for the model to learn their accurate encoding representation, and its retention will introduce a larger error. As a result, we discard the apps with the top 10% and the bottom 10% of the frequency, so as to avoid these apps from adversely affecting the model.

In addition, the mobile device features also include the data flow information of the app. This feature can reflect the user's usage level of each app as well as his specific behavioral inclination. However, different types of apps have different average levels of data flow. For example, video apps consume significantly more data flow than text-reading apps. Even if users spend more time on text-reading apps, their data flow is lower than that of video apps, which users spend less time on. Therefore, if we want to accurately reflect the user's concentration or frequency of using the app, we need to convert the data flow feature. In this paper, we propose the following formula to transform the data flow feature of app:

$$w_{i,j} = \frac{\sum_{k\in\Omega} \frac{|M_k|}{|Z_k|}}{\sum_{k\in\Omega} m_{k,j}} \cdot \frac{|Z_i|}{|M_i|} \cdot m_{i,j} \tag{1}$$

where $w_{i,j}$ represents the level of usage of the i-th app installed by the user u_j, M_i represents the total monthly data flow of the i-th app on all users, Z_i represents the number of monthly live users of the i-th app, $m_{i,j}$ represents the monthly data flow of the user u_j on the i-th app, and Ω represents the set of all apps installed by the user u_j. The meaning of this formula is to compare the proportion of data flow spent by a user on one app with the proportion of the per capita data flow on all users of this app, so as to portray how much the user focuses on this app relative to the whole group. By using this formula, the comparison between apps is transformed into a comparison between users, resolving the issue that the data flow of different types of apps cannot be compared. From this point of view, $w_{i,j}$ actually depicts the relative data flow of the user on the app, which can better reflect the time duration users spend on an app than the data flow.

2) PREPROCESSING OF THE SURVEY DATASET

As mentioned before, the survey dataset consists of questions, corresponding options, and user answers. What we need to do is to encode the user answer data. Since the survey questions allow for both single and multiple choice, users may choose one or more options in a given question, so this paper adopts the multi-hot method to align and encode them. For the i-th question, assuming that it has g_i options, the encoded result of the user's answer to this question will be g_i binary features, expressed as:

$$\boldsymbol{h}_i = \underbrace{(0, \ldots 1, 0, \ldots 1, 0, \ldots)}_{g_i} \tag{2}$$

where the dimension value of 1 indicates that the user has chosen the corresponding option, and the dimension value of 0 indicates that the user has not selected the corresponding option. For example, if a question has four options, A, B, C and D, and one user chooses B, this user's answer code for this question is 0100. Similarly, if he chooses A and C, the result is encoded as 1010. According to this coding

method, the user's answers can be encoded, and then the encoding results of each question are concatenated to obtain the encoding result of the entirely questionnaire, which is expressed as $\boldsymbol{H} = [\boldsymbol{h}_1; \boldsymbol{h}_2; \ldots \boldsymbol{h}_I]$.

C. DATA DRIVEN MODELING

The goal of this paper is to perform user demographic prediction for the users in the mobile dataset. According to the introduction in Section I and Section II, the methods used in previous studies can only make predictions based on single-source data, or deal with multi-source data in a hard-matching method. Therefore, this paper proposes a framework for demographic prediction based on mobile and survey data, as well as a soft-matching method that can fuse multi-source datasets. The main process of our method is to obtain the target demographics of some users in the mobile dataset by means of a survey questionnaire, and then use the survey dataset as an additional information source to extend the target demographics of these survey participants to all massive users of the mobile dataset. However, due to the enormous cost of user surveys, we cannot conduct a one-by-one survey of all mobile users to directly obtain their target demographics. Therefore, there are two main challenges in achieving our goal. To begin, we need to design a data fusion model which could fuse the two heterogeneous datasets of mobile and survey data by comparing the similarity of features across users in these two datasets. Second, we need to extend the target labels from select survey participants to all mobile users based on the results of data fusion. The following parts outline how we achieved these two main key points.

1) MATCHING MODULE

To address the first challenge, we need to determine the similarity between users by comparing their features. According to the description in Section III, features can be divided into directly related features and non-directly related features. As for directly associated features, we use the matching module for direct matching.

As shown in Fig. 4, the role of the matching module is to perform coarse filtering on the survey users and filter out a portion of similar users for the training of the downstream data fusion model. For a user v_i in the survey dataset $V\{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots \boldsymbol{v}_q\}$, its directly related features are $(f_{a,1}^{v_i}, f_{a,2}^{v_i}, \ldots)$. We can match these features with the completely corresponding features $(f_{a,1}^{u}, f_{a,2}^{u}, \ldots)$ in the mobile dataset, and filter out all the users that exactly match the user v_i on these features into a set $U_{match}^{v_i} = \{u_1^{v_i}, u_2^{v_i}, \ldots\}$, and then perform the same operation on all survey users. After merging all the obtained user sets, we can get the matching user set $U_{match} = \{U_{match}^{v_i}\}_{i=1}^{q}$.

The matching module can filter out users who are significantly different from the survey participants. These users do not need to use the model to train and identify. If they enter the training set of the data fusion model, on the one hand, it will consume a lot of resources, and on the other hand, it will make a lot of iterative steps in training extremely simple rules, which would be detrimental to the convergence of the fusion model. Therefore, the significance of the matching module is to relieve the pressure on the training resources of the data fusion model and accelerate the convergence of the model at the same time.

2) DEEP STRUCTURED FUSION MODEL (DSFM)

After comparing the directly related features, we should handle the non-directly related features. Unlike the directly related features, these features are not exactly the same in mobile and survey data and cannot be compared directly. When designing the questionnaire, the reason why the questions are not completely corresponding to the features of mobile data is to ensure the model's generalization. Our purpose is to extend some user demographics in survey data to all mobile users. If these user profiles are too detailed, it will lead to the convergence of the model and the performance of the overall fusion will deteriorate, so the existence of non-directly related features is meaningful. For non-directly related features, it is difficult to manually design rigorous rules for processing, hence a data fusion model must be used to handle them. What the data fusion model needs to do is to learn representations of the mobile and survey data, and map users of the two datasets from different feature spaces to the same. In this way, user similarity can be compared by processing user representation vectors.

In order to achieve this, this paper proposes a data fusion model named Deep Structure Fusion Model (DSFM). DSFM is made up of neural networks with attention mechanisms, and it achieves the goal of data fusion by transferring it into a binary classification problem. The model architecture is shown in Fig. 5. The input of the model is all the non-directly related features of a pair of users from mobile and survey data after the preprocessing operation introduced in Section IV, and the output is the similarity score of this pair of users. This model mainly includes the following components:

Embedding layer. The main role of the embedding layer is to map ID features such as apps from sparse one-hot encoding to dense vectors with low dimension. The ID features not only contain the name of used apps, but also the category of them. The name and category of the same app are encoded by the embedding layer and then assembled together using a concatenating layer to form a vector for subsequent processing. The reason for this is that there are many long-tail apps, and adding category features is equivalent to transforming the feature granularity from fine to coarse, which can alleviate the impact of data sparsity to a certain extent [55]. The data flow feature introduced in Section IV will be added to the embedding layer as the weight of each app. In addition, the model not only embeds app features but also other features of mobile data, including communication features and other mobile device features. The dimension of the vectors after embedding can be determined according to the number of feature categories but the vectors which need to do attention computations must have the same dimension.

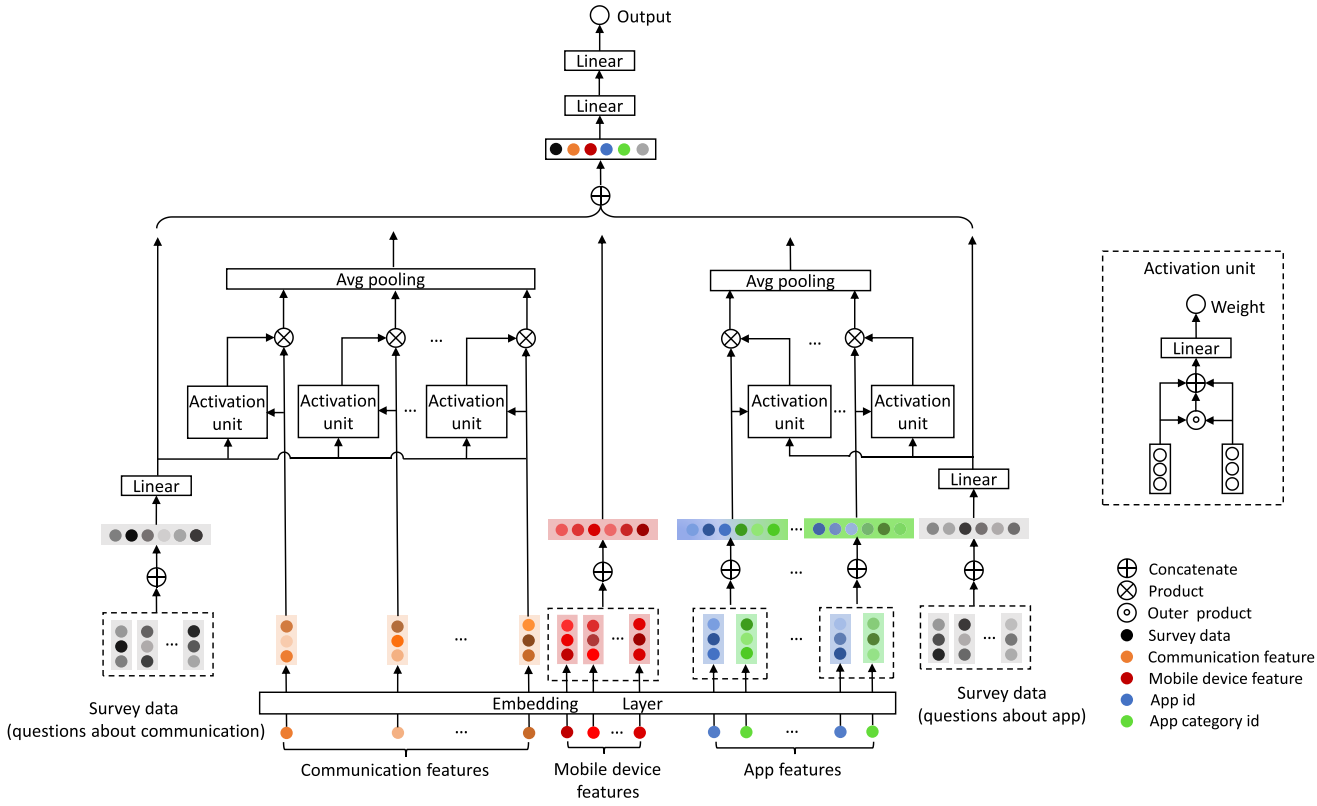


FIGURE 5. The structure of the deep structured fusion model (DSFM).

Concatenation layer. The concatenation layer concatenates multiple input vectors $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots \boldsymbol{e}_k\}$ in the given dimension into one vector, which is convenient for subsequent network layers to perform overall processing and deep interaction of features. The size of the output vector in the concatenating dimension is the sum of all the original vectors, as shown in the following formula:

$$out_{cat} = concat(\{\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots \boldsymbol{e}_k\}) = [\boldsymbol{e}_1; \boldsymbol{e}_2; \ldots \boldsymbol{e}_k] \quad (3)$$

Pooling layer. The pooling layer also combines multiple input vectors into one output. Different from the concatenation layer, the pooling layer in DSFM uses average pooling, that is, all vectors are summed and divided by the number of vectors. This layer requires that the size of all input vectors must be the same, and the size of the output vector is also the same as the input vector, as shown in the following formula:

$$out_{pool} = avgpooling(\{\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots \boldsymbol{e}_k\}) = \frac{1}{k} \sum_{i=1}^{k} \boldsymbol{e}_k \quad (4)$$

In DSFM, the pooling layer mainly acts on the weighted results of app features and communication features after activation. In the process of pooling, the initial interaction and fusion between vectors have been completed.

Activation unit. The activation unit in DSFM works by determining the degree of correlation between the two input feature vectors, and then deciding how much the current vector will affect the model output. For example, in an input sample of the current survey data, the user has selected game apps in the question "What types of apps do you currently have membership in?", which means that this user is probably a game lover. Among the app features of the mobile data, game-related apps should receive higher attention from the model, so they are given higher weights via the activation units. We utilize attention mechanisms in the activation unit to accomplish this task. Since the attention mechanism of the neural network [56] was proposed, it has been widely used in various scenarios, such as natural language understanding, image recognition, recommender systems, etc. The attention mechanism enables the neural network model to locate key information from complex data and features, thereby improving training efficiency and model performance. The activation unit is shown in Fig. 5, and adopts the similar structure as the activation unit in the DIN model [57]. The unit takes two vectors of the same size as input and calculates their outer product. The outer product result is concatenated with the input vectors to yield the final output after passing through a fully connected layer. This output is the attention weights of the two input vectors. After getting the attention weight, we can multiply the original vector by it, and then input the result into the average pooling layer to interact with other weighted vectors, as shown in the following formula, where

o represents the outer product calculation:

$$out_{ac-unit} = \frac{1}{k}\sum_{i=1}^{k} o(\boldsymbol{e_i}, \boldsymbol{v}) * \boldsymbol{e_i} \tag{5}$$

In DSFM, the attention mechanism is mainly used for local activation of app features and communication features. The communication features correspond to the communication questions in the questionnaire of survey data, and the app features correspond to the app questions. These two types of features have a clear correlation with the survey data. Except for the app features, the remaining mobile device features do not need to be processed by the attention mechanism due to the weak correlation with the survey data.

Linear layer. The linear layer, also known as the fully connected layer, is responsible for implementing feature interaction and transforming the input vector's dimension. The linear layers in DSFM all use the activation function of Leaky ReLU [58] except for the output layer, as shown below:

$$f(x) = \max(ax, x) = \begin{cases} ax, & if\ x < 0 \\ x, & if\ x \geq 0 \end{cases} \tag{6}$$

Compared with the ReLU function, it solves the zero-gradient problem of negative values and is able to retain more information.

The layers introduced above are the main internal structure of DSFM. In addition, some common layers in neural networks, such as the dropout layer and the normalization layer, are also used in DSFM and play an important role in model performance, but they are not shown in Fig. 5.

On the whole, the input of DSFM is a user pair, one from the survey dataset and the other from the mobile dataset. In this way, we can see that not only the survey users can be input into the model, but the non-survey users in the mobile dataset can also be trained by DSFM. After the samples are fed into the model, the mobile data features are encoded through the embedding layer. Following encoding, the vectors of the communication features in mobile data and the communication questions in survey data are used for attention calculation to obtain the feature weight, and then passed into the average pooling layer to achieve weighted summation. The app features are subjected to the same operation as the app questions in survey data. After that, the five vectors, including the vector of communication questions, the vector of app questions, the vector of mobile device features, and the weighted vectors of communication and app features in mobile data are concatenated together to obtain the final results through the full connection layer. According the description in section III, since the goal of the fusion model is to judge whether the user entities from two heterogeneous datasets are the same person in reality, we model the data fusion problem as a binary classification problem in the process of DSFM training, and use the cross-entropy loss function to calculate the loss between the predicted result $p(\boldsymbol{x})$ and the true label y:

$$Loss = -\frac{1}{N}\sum_{(\boldsymbol{x},y)} (y * \log p(\boldsymbol{x}) + (1-y) * \log(1-p(\boldsymbol{x}))) \tag{7}$$

Among them, N is the total number of samples, $\boldsymbol{x}$ is the input of the model, $p(\boldsymbol{x})$ is the predicted value of the model, y is the true label and its value set is $\{0, 1\}$. 0 means the current input survey user is not the same person as the input mobile user, whereas 1 means the same person. Depending on the adjustment of the ratio of positive and negative samples, we can generate training samples which are thousands of times larger than survey users. This means that our model will not be restricted to survey users and can be well trained with enough training samples. After training using the cross-entropy loss function, we can regard the final prediction result as the similarity score between two input users. In this way, this fusion model can be used to determine the similarity between the pair of users from two heterogeneous datasets, and realize the soft matching of them.

The reason why we use neural networks to build the fusion model is to take full advantage of the app features. Among the user features to be processed, app features in mobile data are actually a type of ID feature with sparse values and a long feature encoding length. The best way to deal with these ID features is to embed and map them into low-dimensional dense features, which can be accomplished using neural network's embedding layer. App features can accurately reflect the personalized behavior characteristics and patterns of people [59], [60], [61], which is highly helpful for use similarity comparison and data fusion, thus the survey questionnaire also includes several questions related to app usage. In this case, the usage of data fusion model based on neural networks outperforms other traditional machine learning algorithms such as support vector machine and decision trees in terms of representation learning.

3) USER DEMOGRAPHIC PREDICTION

After obtaining the data fusion model, we need to address the second challenge that is extending the target demographics from some survey participants to all massive users of the mobile dataset. As illustrated in Fig. 4, we can use the similarity score output by the fusion model to achieve this. There are many ways to enhance the prediction algorithms' performance, such as alpha-beta filter and deep extreme learning machine [62]. However, according to the specific form of DSFM, we propose two strategies for predicting demographics based on user similarities. The first is the weighted sum method. We use DSFM to compare current user's similarity $\{s_1, s_2, \ldots s_q\}$ to all q survey participants, and then divide each similarity score by the sum of them to get $\{\bar{s}_1, \bar{s}_2, \ldots \bar{s}_q\}$ where $\bar{s}_1 + \bar{s}_2 + \ldots \bar{s}_q = 1$. We take them as the weights to multiply the target demographic value of the corresponding survey participants $\{r_1, r_2, \ldots r_q\}$, and then sum up the whole result as current user's target demographic value R_{cur}, as

shown in the following formula:

$$R_{cur} = \sum_{i=1}^{q} r_i * \bar{s}_i \tag{8}$$

It should be noted that, if the proportion of positive and negative samples in the training set used for DSFM training differs from the current dataset, then the similarity score needs to be corrected. Referring to the method of correcting the user click through rate in the CTR model, the user similarity score needs to be corrected according to the following formula [63]:

$$s_i^{new} = \frac{s_i}{s_i + \frac{1-s_i}{c}} \tag{9}$$

the sampling rate c is determined based on the quotient of the positive and negative sample ratio in the current dataset and the model training set.

The second strategy to obtain the current user's demographics based on the similarity score is the proportion difference method. First, sort the survey participants according to their similarity score with the current user, and then calculate the proportion of users with each value of the demographic among the top 10% users and overall users. We can subtract the proportion of user number with each value in the two groups, and the value with the largest difference can be regarded as the current user's target demographic. For example, for the gender prediction of the current user, there are a total of 1,000 survey users, 70% males and 30% females. Among the top 10% of users with similarity, the proportion of males and females is 60% and 40%. After subtracted by the overall ratio, the difference is -10% and 10%, then 10% is a larger value, so the gender of the current user is judged to be female. It can be found that although there are more males in the top 10% of users, since the overall proportion is that males are larger than females, the final judgment must remove the influence of the imbalance between the two classes of the sample. Compared with the first strategy, this method does not require an accurate similarity score, and can deal with the negative impact caused by the unbalanced number of categories. The disadvantage is that the numerical sorting is required. When the number of survey users is relatively large, it will be more time-consuming. In practice, the choice between these two strategies can be made based on the circumstances.

V. EXPERIMENT EVALUATION

In this section, we validate the effectiveness of DSFM proposed in this paper for user demographic prediction using the two datasets introduced in Section IV. The mobile dataset contains almost one billion users, while the survey dataset contains 29,809 users, both of whom were collected in June 2022. We select the gender and age characteristics of mobile users as the demographics to be predicted, which are also the targets chosen by many demographic prediction studies. They are very important for the construction of user profiles, but in real life, they are frequently difficult to obtain due to privacy concerns and other factors [34]. As a result, the prediction of such features is of great practical significance. To train and test the model performance, we delete the gender and age features in mobile data in order to predict them. However, we retain the gender and age questions in survey data for DSFM to get demographics of sample survey participants and then extend them to all mobile users. We compare the effect of DSFM with other baseline models, examine the model's generalization, and perform ablation experiments on various feature combinations of the datasets.

A. EVALUATION METRICS

This experiment focuses on the prediction of user gender and age. User gender prediction is a typical binary classification problem, whereas general age prediction is a regression problem, because age is a continuous variable. However, in order to simplify the experimental setting, we divide the user's age into several buckets to turn the regression problem into a classification problem. There are many kinds of bucketing methods in age prediction task, and we select a traditional approach [45] as the main task for age prediction. The method divides the ages into five buckets, namely Matures, Baby Boomers, Gen X, Gen Y and Gen Z. However, since the experimental data in this paper are from the mobile dataset of operators, the users' ages are concentrated between 18 and 70 years old, so we discard the Matures group, and at the same time make some corrections to the age groups of other buckets, as shown in Table 5. In this way, the age prediction problem is transformed into a four-class problem.

For gender prediction, since it is a binary classification problem, we use precision, recall, F1- score, AUC and accuracy as evaluation metrics. But for age prediction, since we model it as a four-class problem, and metrics such as precision, recall, f1, and AUC can only be used to measure binary classification problems, we use the accuracy as the evaluation metric for multi-class problem.

For binary classification problems, precision indicates the proportion of true positives among all predicted positives. Recall indicates the proportion of true positives among the predicted values of all positive samples. F1-score is the combined result of precision and recall. Accuracy indicates the proportion of the number of correctly classified samples among the total number of samples. Besides these metrics, we use AUC to measure the comprehensive performance of the model. AUC represents the area enclosed by the ROC curve and the x-axis, and also indicates the probability that a randomly chosen positive sample is ranked higher than a randomly chosen negative sample [64]. The calculation formulas of these metrics are as follows, where TP, FP, TN, and FN represent true positives, false positives, true negatives and false negatives in the confusion matrix respectively, as shown in Table 6.

$$precision = \frac{TP}{TP + FP} \tag{10}$$

$$recall = \frac{TP}{TP + FN} \tag{11}$$

TABLE 5. Buckets in the main task of age prediction.

Category	Birth Year	Age
0 (Gen Z)	1995-2004	18-27
1 (Gen Y)	1980-1994	28-42
2 (Gen X)	1965-1979	43-57
3 (Baby Boomers)	1944-1964	58-78

$$F_1 - score = \frac{2 * precision * recall}{precision + recall} \tag{12}$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{13}$$

B. BASELINE METHODS

We compare the performance of DSFM with traditional machine learning and ordinary deep learning models. We select logistic regression, support vector machine, decision trees and artificial neural networks as the baseline models, which are also commonly used to solve classification problems.

1) LOGISTIC REGRESSION (LR)

Logistic Regression is a generalized linear classifier based on maximum likelihood estimation, which uses the sigmoid function to constrain the output of the model between 0 and 1. Logistic regression can be used directly for binary classification problems, whereas for multi-classification problems, a one-vs-one or one-vs-all approach is required. Considering that the one-vs-all method tends to case an imbalance in the amount of data between different classes, and its actual effect is worse than one-vs-one, so we choose the one-vs-one method for unified processing. Take the age prediction to be done in this paper as an example, which will be modeled as a four-class classification problem. Assuming that the age labels are grouped as $\{0, 1, 2, 3\}$, we need to train six logistic regression models which predict whether the label is 0 or 1, 0 or 2, 0 or 3, 1 or 2, 1 or 3 and 2 or 3 respectively. The final result is voted based on the prediction results of these six classifiers.

2) SUPPORT VECTOR MACHINE (SVM)

SVM is a classifier based on the maximum interval classification hyperplane [65]. Similar to logistic regression, the support vector machine can also directly predict the binary classification problem and needs the one-vs-one approach for multi-classification problems. In this experiment, we use the SVM model based on the polynomial kernel function and Gaussian kernel function to predict the demographics.

TABLE 6. The confusion matrix of gender prediction.

	Male	Female
Male	TP	FP
Female	FN	TN

3) LIGHT GRADIENT BOOSTING MACHINE (LightGBM)

This is a framework for implementing the decision tree algorithm [66]. This model uses the binary decision tree as a weak classifier and is trained by repeated iteration and gradient boosting to get better performance. LightGBM can realize the random forests (RF) and gradient boosting decision tree (GBDT) algorithms which can be directly used in machine learning problems such as binary classification, multi-class classification and regression.

4) ARTIFICIAL NEURAL NETWORK (ANN)

To compare the performance of DSFM to ordinary deep learning models, we employ the commonly used ANN models for direct gender and age prediction. The model structure is shown in Fig. 6, including embedding layer, concatenation layer and linear layer. The function of each layer is similar to that in DSFM. The expressive ability of the model can be altered by adjusting the number of the linear layer. Unlike DSFM, ANN removes the module for processing survey data and can only make demographic prediction based on single-source data.

In our experiment, we apply these baseline models to the mobile dataset without adding the survey data. Because these models do not have the ability to fuse multi-source heterogeneous datasets, we directly input the features of mobile data into the baseline models and predict users' gender and age, respectively. For ID features such as app names and categories, traditional machine learning models cannot use the embedding layer for end-to-end processing like the models based on neural networks such as ANN and DSFM. But the one-hot encoding feature's dimension is too large for the baseline models to handle, so we utilize the Item2vec method [67] to reduce the dimensionality of the name list and category list of used apps before the baseline models' training.

C. EVALUATION RESULTS

We train the baseline models and DSFM on the experimental datasets. After completing the data preprocessing operation, we perform the data modeling as previously described. First, the matching module is used to filter users from the massive mobile data according to the directly related features of the

TABLE 7. Accuracy on a real-world mobile dataset and comparison with baseline methods.

Method		Accuracy for Gender	Accuracy for Age
LR		0.6352	0.4755
SVM	Polynomial Kernel	0.6298	0.4676
	Gaussian Kernel	0.6503	0.5136
LightGBM	Random Forest (RF)	0.7223	0.6104
	Gradient Boosting Decision Tree (GBDT)	0.7493	0.6277
ANN	Three Linear Layers	0.7456	0.6358
	Five Linear Layers	0.7212	0.6275
DSFM	Weighted Sum Method	**0.7781**	**0.6630**
	Proportion Difference Method	**0.7816**	**0.6879**

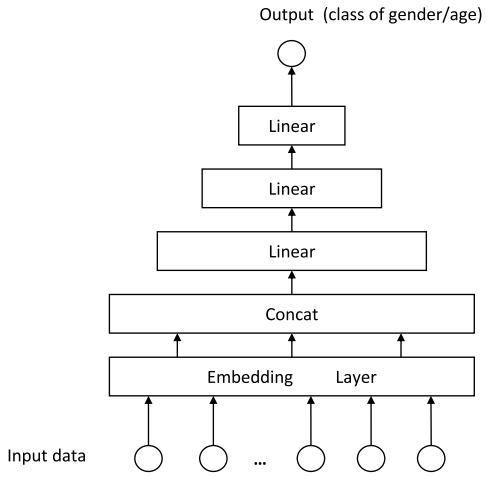


FIGURE 6. The structure of the artificial neural network.

survey data. In this experiment, we screen out 208,271 recall users based on 29,809 survey participants for the training of DSFM. In addition, we randomly select 100,000 users from the mobile dataset except for these 208,271 users as the test set. The baseline models can be trained directly based on the training set composed of these 208,271 users. When training DSFM, it is also necessary to introduce the survey data. At this time, the model is input in the form of pairwise, one user pair at a time, including a survey user and a mobile user. The target value is whether these two users are the same person. The 29,809 survey participants have been associated with the users in the mobile dataset in advance by means of user ID. The model uses the Leaky ReLU activation function for all hidden layers but the output layer uses the sigmoid function. ANN and DSFM use cross-entropy to calculate the loss and an Adam optimizer for optimization. The learning rate of DSFM is set to 0.0006, and 5,000,000 iterations are performed. According to the AUC and loss of the model, the performance changes are monitored in real time to determine whether the training can be stopped. All models undergo 5-fold cross-validation during training, and after training is done, we evaluate the performance of all models on the test set.

1) PERFORMANCE COMPARISON

After training with 5-fold cross-validation, the performance results of the baseline models and DSFM on the test set are shown in Table 7. It can be seen that DSFM has achieved the best performance, whether the final result is calculated using the weighted sum method or the proportional difference method. The accuracy of gender and age prediction using DSFM is higher than all baseline models. At the same time, compared with the best baseline results, the accuracy of gender prediction is increased by 0.0323, and the accuracy of age prediction is increased by 0.0521. For further verification, we measure the precision, recall, F1-score and AUC of all models above the binary classification problem of gender prediction, as shown in Table 8. For all baseline models, we choose the hyperparameters and conditions that would make them perform best. It is obvious that the DSFM proposed in this paper can achieve significantly superior results in all metrics, thus proving the effectiveness of our proposed demographic prediction method based on the fusion of mobile and survey data.

From Table 7, the SVM with Gaussian kernel outperforms the polynomial kernel, LightGBM with GBDT algorithm outperforms the RF algorithm. These findings are generally consistent with common experience. It is worth mentioning that the three-layer ANN performs better than the five-layer ANN. Although the neural network model with more linear layers has stronger expression ability, its training difficulty will also increase. If the data volumes cannot meet the training needs of the network, then blindly increasing the number of layers will reduce the performance of ANN.

Among the baseline models, the LightGBM model with GBDT performs the best in gender prediction and ANN with three linear layers perform the best in age prediction. LightGBM and ANN significantly outperforms the LR and SVM models, which is because the model expressive ability

TABLE 8. Model performance for gender prediction.

Method	Precision	Recall	F_1-score	AUC	Accuracy
LR	0.6533	0.6282	0.6405	0.6118	0.6352
SVM using Gaussian Kernel	0.6573	0.6940	0.6752	0.6265	0.6503
LightGBM using GBDT	0.7331	0.7758	0.7538	0.7203	0.7493
ANN using three linear layers	0.7421	0.7263	0.7341	0.7135	0.7456
DSFM using proportion difference method	**0.8016**	**0.7798**	**0.7905**	**0.7723**	**0.7816**

of LR and SVM are naturally weaker than decision trees and neural networks. In addition, the LR and SVM models can only model binary classification problems. Although we use the one-vs-one method to extend it, we can see in Table 7 that the effects of these two models for the multi-class classification problem decrease more compared to the binary classification problem, which is limited by model's principle and structure. It can be seen that in the traditional machine learning models, the models based on decision trees can often be applied to a wider range of scenarios and achieve better performance at the same time. Besides that, the difference in performance between LightGBM and ANN is not significant in Table 7 and Table 8. Both models are highly expressive, with LightGBM being slightly better for gender prediction problem, and ANN for the age prediction problem.

As for the DSFM proposed in this paper, its prediction accuracy on the two demographics can also be significantly higher than the accuracy of the best baseline model. When predicting demographics based on the user similarity given by DSFM, the proportional difference method can achieve better performance. As can be seen from Table 7, compared with the weighted sum method, the proportional difference method achieves an improvement of 0.0035 for gender prediction, but an improvement of 0.0249 can be achieved for age prediction, which is more obvious. This is due to the fact that compared with the proportional difference method, the weighted sum method needs to use the precise values of the similarity scores of output by DSFM for calculation. The proportional difference method only needs to employ these similarity scores for sorting, and focus on the relative magnitudes rather than absolute values. Since DSFM mainly monitors the model performance's changes through AUC during training, it only focuses on the sorting ability of the model for positive and negative samples. Therefore, there is a certain fluctuation in the specific value of the similarity scores output by DSFM. Although the corresponding correction has been made, it will still have a certain impact on the demographic prediction. The accuracy requirement for the output of DSFM increases for the four-class classification problems such as age, because the values of the category are greater. This is why the improvement of the proportional difference method over the weighted sum method is more obvious for age prediction. Since DSFM using the proportional difference method performs better, the remaining analysis of DSFM in this paper is based on the proportional difference method.

2) MODEL GENERALIZATION

In order to measure the model generalization, we evaluate the impact of the number of survey participants on the model performance. Fig. 7 depicts the accuracy change of gender prediction, whereas Fig. 8 depicts the accuracy change of age prediction. We test the DSFM's accuracy change for demographic prediction when the number of survey participants is reduced from 29,809 to 20,000, 15,000, 10,000, 5,000, and 1,000 by random sampling. All models undergo 5-fold cross-validation during training. It can be seen from Fig. 7 and Fig. 8 that with the decline in the number of survey participants, the model performance shows a decreasing trend for both age and gender prediction, which is inevitable. The decline in the number of survey participants means that both the number of training samples and true labels provided to DSFM are decreasing, and the more they decrease, the more difficult it is for the model to accurately learn the criteria for user similarity. As a result, prediction accuracy will deteriorate.

Although the decline in the number of survey participants will lead to a decrease in model performance, even when the number of survey participants is only 1,000, which means the number of users has dropped by more than 95%, the accuracy of demographic prediction using DSFM is still higher than the best accuracy of baseline models. It proves that using DSFM for demographic prediction based on multi-source heterogeneous datasets is better than using only a single dataset. The survey dataset adds external information as an additional data source, and DSFM proposed in this paper makes the fusion of multi-source data possible. The combination of these two factors can achieve better results than traditional methods.

In addition, two extreme cases can be inferred from the trends in Fig. 7 and Fig. 8. When the number of survey participants drops to 0, which means there is no survey data at all and

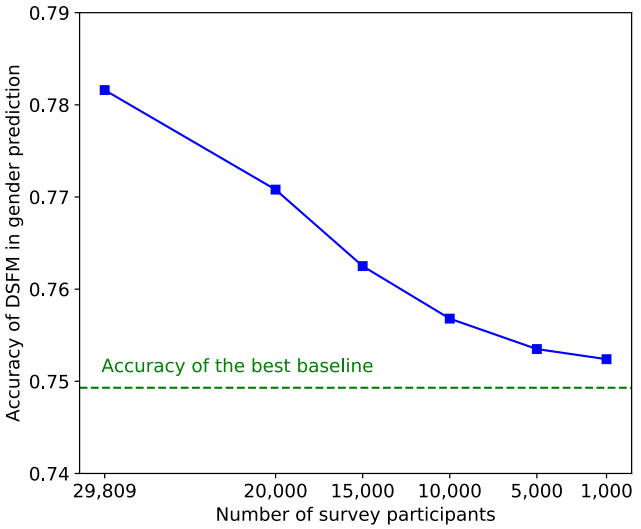


FIGURE 7. The impact of the number of survey participants to the gender prediction accuracy with DSFM.

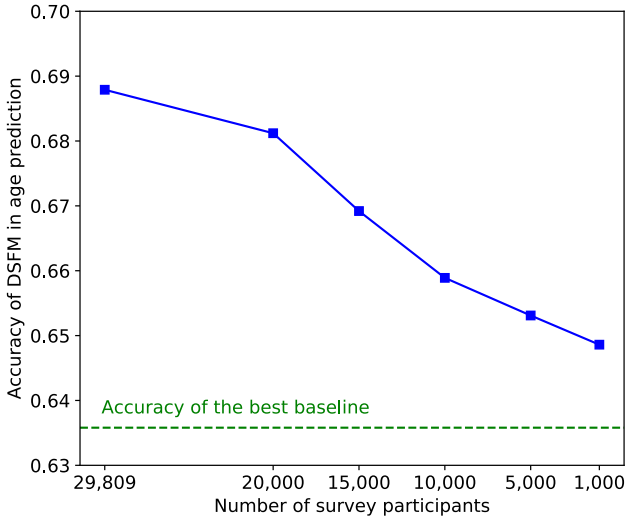


FIGURE 8. The impact of the number of survey participants to the age prediction accuracy with DSFM.

we predict user demographics based only on mobile big data using a deep learning model. At this point, the performance is completely controlled by the model itself. This demonstrates that if we only use deep models to predict user demographics on mobile data without survey data, the accuracy will be lower than that we have achieved using DSFM and data fusion. This inference echoes that the accuracy of ANN in Table 7 is significantly lower than that of DSFM. On the contrary, when the number of survey participants increases to the same level as mobile users, it means that almost one billion users have been surveyed. At this time, in theory, we can directly obtain the target demographics of all users. We do not even need to use the fusion model to compare the similarity, and the prediction accuracy will be 100%. But this is obviously unrealistic. Researching such a huge number of users will consume a lot of manpower and material resources, and it is not a reasonable and scientific research method. Therefore, the quantity of survey participants we choose needs to be set in an appropriate range. If it is too small, the model's performance will be degraded, and if it is too enormous, it will be prohibitively expensive. At this point, the role of DSFM proposed in this paper is reflected. By learning from the mobile data and a certain amount of sample survey data, the target demographic is extended from some survey participants to all massive users, which not only improves the model performance but also saves a lot.

The prior age prediction is to divide the age into four categories and convert it into a four-class classification problem. If the number of age buckets is adjusted, then the model performance will also change. Based on the standard segmentation shown in Table 5, we further merge and split the age buckets in order to test the generalization of DSFM when the number of categories changes. We adjust the number of age buckets according to Table 9, and measure the prediction accuracy of DSFM, LightGBM and ANN when the number of age buckets changes to 3, 6, 8, and 10. The results are shown in Fig. 9. It can be found that with the increase of the number of buckets, the performance of these three models has declined to a certain extent. This is an inevitable situation because it is always easier to perform a three-class classification than a ten-class classification. However, regardless of the number of buckets, the prediction accuracy of DSFM is always higher than that of LightGBM and ANN model. When the number of buckets is 10, the prediction accuracy of DSFM can still remain around 0.40. It can be seen that DSFM can achieve better results even when dealing with multi-class classification problems with a large number of categories.

3) ABLATION STUDY

In this part, we take the gender prediction problem as an example to analyze the model's ablation study. The ablation study in this paper is used to examine the impact of the communication features and mobile device features in mobile data, as well as the communication and app questions in survey data on model performance. Besides that, this ablation study demonstrates that our method can predict demographics that are less relevant to the current dataset's features, which will broaden the application scenarios. The model in the ablation study all undergo 5-fold cross-validation.

We first remove the mobile device features and communication features in mobile data, but preserve all the questions of survey data, and then examine the changes in model performance of DSFM and LightGBM with GBDT in these cases. We choose the LightGBM with GBDT to compare with our model since it can get the best performance in the baseline models. As shown in Fig. 10, when some features are deleted, the performance of the two models is degraded to a certain extent, which indicates that both communication features and mobile device features have some relevance for the user gender. When only the communication features are preserved, the performance of LightGBM decreases more than only the mobile device features are retained, and the accuracy drops

TABLE 9. The details of the age bucket.

Number of age bucket	Details of age groups	Birth year of age groups
3	[18, 27], (27, 57] and (57, 78]	1995-2004; 1965-1994; 1944-1964
4	[18, 27], (27, 42], (42, 57] and (57, 78]	1995-2004(Generation Z); 1980-1994(Generation Y); 1965-1979(Generation X); 1944-1964(Baby Boomers)
6	[18, 27], (27, 37], (37, 47], (47, 57], (57, 67] and (67, 78]	1995-2004; 1985-1994; 1975-1984; 1965-1974; 1955-1964; 1944-1954
8	[18, 27], (27, 32], (32, 37], (37, 47], (47, 57], (57, 62], (62,67] and (67, 78]	1995-2004; 1990-1994; 1984-1989; 1975-1984; 1965-1974; 1960-1964; 1955-1959; 1944-1954
10	(18, 27], (27, 32], (32, 37], (37, 42], (42, 47], (47, 52], (52, 57], (57, 62], (62, 67] and (67, 78]	1995-2004; 1990-1994; 1984-1989; 1980-1984; 1975-1979; 1970-1974; 1965-1969; 1960-1964; 1955-1959; 1944-1954

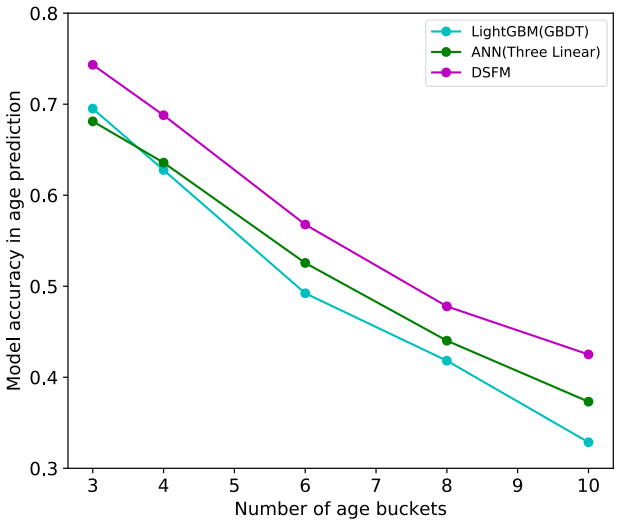


FIGURE 9. The accuracy of LightGBM, ANN and DSFM in age prediction under different bucket conditions.

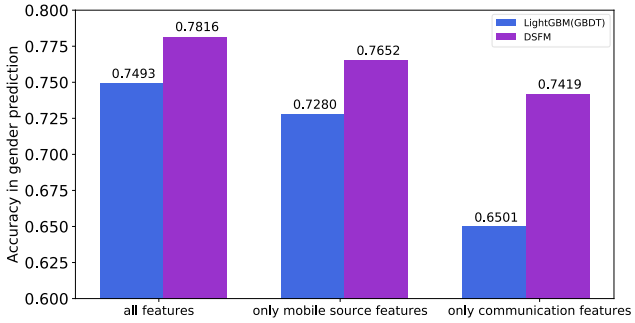


FIGURE 10. The accuracy of DSFM and LightGBM in gender prediction under different feature combinations.

by 0.0992. Since LightGBM only predicts user demographic based on the mobile dataset, this indicates that the correlation between gender and mobile device features is higher, and gender prediction using simply communication features would perform poorly. The same is true for DSFM. The effect of DSFM is better when the mobile device features are preserved. However, it is worth noting that when only the communication features are preserved, the performance of LightGBM suffers more than DSFM, while the accuracy of DSFM only drops by 0.0397. Despite the fact that the previous conclusion proves that the communication features are more related to gender, DSFM nevertheless retains significant accuracy based on these poorly correlated features. This is due to the fact that DSFM compensates for the constraints of the current dataset features by incorporating an external information source. This suggests that when utilizing DSFM for demographic prediction, it is possible to predict demographics that have a low correlation with the current dataset's features. We only need to ensure the quality of the additional dataset, that is, the survey data. In this way, DSFM can predict more kinds for demographics and expand the utilization situation.

In addition to altering the features of the mobile dataset, we go further by adjusting the question composition of the survey data and examining how model performance changes under different conditions. Table 10 displays the specific feature combination details. There are five scenarios in total. It can be found that the first, second, and fourth scenario are all studied in Fig. 10. On this basis, we add two scenarios of removing communication questions or app questions in the survey dataset. Fig. 11 depicts the results of testing the performance of DSFM in these five scenarios. We mainly compare the performance differences between scenarios 2 and 3, and scenarios 4 and 5. It can be found that when the questions corresponding to mobile data in the survey data are deleted, the performance of DSFM will drop significantly, because the correlation between these two datasets is destroyed at this time. In scenario 4, according to the previous analysis, although the mobile device features in mobile data which are more related to gender are erased, and only the communication features are preserved, the performance of DSFM

TABLE 10. Different situations of feature combination.

	Communication features of mobile data	Mobile source features of mobile data	Communication questions of survey data	App questions of survey data
1	√	√	√	√
2		√	√	√
3		√	√	
4	√		√	√
5	√			√

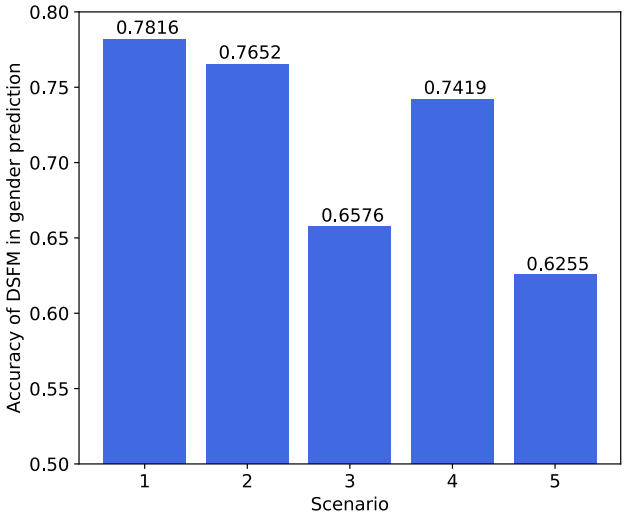


FIGURE 11. The accuracy of DSFM in gender prediction under different scenarios.

remains good and does not decline much. This is due to the fact that mobile and survey data can still be correlated based on communication features and the communication questions in survey data, hence DSFM incorporates external information from survey data to compensate for the lack of information to some extent. However, in scenario 5, removing the communication questions from the survey data destroys the link between the mobile and survey data. In this case, DS-FM is unable to gather sufficient information from the survey data, and model performance suffers dramatically. This demonstrates that although DSFM can predict demographics with low relevance to the current mobile dataset, it must be ensured that there is a certain correlation between the mobile and survey data, so that the model can obtain the information related to the demographics from the survey data, otherwise the model accuracy will be significantly reduced. After all, it cannot create something out of nothing.

According to the experimental results in this section, DSFM proposed in this paper significantly outperforms the traditional demographic prediction models in performance. The advantage of this model is that it can improve prediction accuracy by incorporating external data sources, and at the same time, it can predict labels that are less relevant to the features of the current dataset when certain conditions are met. As a result, it has the potential to broaden the scope of demographic prediction.

VI. CONCLUSION

In this paper, we propose a framework for predicting user demographics based on multi-source data and design a model named Deep Structure Fusion Model (DSFM) that enables data fusion between two heterogeneous datasets in a soft matching method by comparing user similarity. The framework proposed in this paper significantly improves the accuracy of user demographic prediction results by incorporating an external data source and performing data fusion. At the same time, our framework expands the scope of the demographic to be predicted, so that it is no longer completely limited by the features of the current dataset. We test the performance of the framework on an anonymous real-world mobile dataset, and it achieves state-of-the-art results in gender and age prediction, with an accuracy of 0.7816 and 0.6879. The framework proposed in this paper has good practicability and portability, and can be used for user demographic prediction for various types of datasets in multiple scenarios. In the future, we will continue to study the use of natural language understanding models such as BERT to automate the processing of questionnaire data and the attempt of contrastive learning to overcome the few-shot learning problem that may arise during the data fusion process.

REFERENCES

[1] H. Aksu, L. Babun, M. Conti, G. Tolomei, and A. S. Uluagac, "Advertising in the IoT era: Vision and challenges," *IEEE Commun. Mag.*, vol. 56, no. 11, pp. 138–144, Nov. 2018.

[2] X. Qin and Z. Jiang, "The impact of AI on the advertising process: The Chinese experience," *J. Advertising*, vol. 48, no. 4, pp. 338–346, Aug. 2019.

[3] B. J. Jansen, K. Moore, and S. Carman, "Evaluating the performance of demographic targeting using gender in sponsored search," *Inf. Process. Manage.*, vol. 49, no. 1, pp. 286–302, Jan. 2013.

[4] G. Zhou, N. Mou, Y. Fan, Q. Pi, W. Bian, and C. Zhou, "Deep interest evolution network for click-through rate prediction," in *Proc. AAAI Conf. Artif. Intell.*, Jul. 2019, vol. 33, no. 1, pp. 5941–5948.

[5] X.-R. Sheng, L. Zhao, G. Zhou, X. Ding, and B. Dai, "One model to serve all: Star topology adaptive recommender for multi-domain CTR prediction," in *Proc. 30th ACM Int. Conf. Inf. Knowl. Manag.*, 2021, pp. 4104–4113.

[6] S. Hu, A. Kumar, F. Al-Turjman, S. Gupta, and S. Seth, "Reviewer credibility and sentiment analysis based user profile modelling for online product recommendation," *IEEE Access*, vol. 8, pp. 26172–26189, 2020.

[7] H.-T. Cheng, L. Koc, J. Harmsen, and T. Shaked, "Wide & deep learning for recommender systems," in *Proc. 1st Workshop Deep Learn. Recommender Syst.*, 2016, pp. 7–10.

[8] H. Guo, R. Tang, Y. Ye, Z. Li, and X. He, "DeepFM: A factorization-machine based neural network for CTR prediction," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Aug. 2017, pp. 1725–1731.

[9] X. Chen, P. Zhao, J. Xu, Z. Li, L. Zhao, and Y. Liu, "Exploiting visual contents in posters and still frames for movie recommendation," *IEEE Access*, vol. 6, pp. 68874–68881, 2018.

[10] R. Rossi, M. Gastaldi, and F. Orsini, "How to drive passenger airport experience: A decision support system based on user profile," *IET Intell. Transp. Syst.*, vol. 12, no. 4, pp. 301–308, May 2018.

[11] A. Padró-Arocas, D. Mena-Tudela, E. Baladía, A. Cervera-Gasch, V. M. González-Chordá, and L. Aguilar-Camprubí, "Telelactation with a mobile app: User profile and most common queries," *Breastfeeding Med.*, vol. 16, no. 4, pp. 338–345, Apr. 2021.

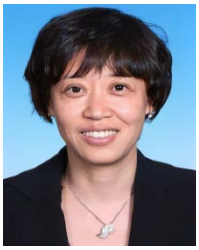
[12] I. Kim and G. Pant, "Predicting web site audience demographics using content and design cues," *Inf. Manage.*, vol. 56, no. 5, pp. 718–730, Jul. 2019.
[13] O. Dogan and B. Oztaysi, "Genders prediction from indoor customer paths by levenshtein-based fuzzy kNN," *Expert Syst. Appl.*, vol. 136, pp. 42–49, Dec. 2019.
[14] A. Bwambale, C. F. Choudhury, and S. Hess, "Modelling trip generation using mobile phone data: A latent demographics approach," *J. Transp. Geography*, vol. 76, pp. 276–286, Apr. 2019.
[15] A. I. Al-Ghadir and A. M. Azmi, "A study of Arabic social media users—Posting behavior and author's gender prediction," *Cogn. Comput.*, vol. 11, no. 1, pp. 71–86, Sep. 2018.
[16] D. Nguyen, D. Trieschnigg, and A. S. Dogruöz, "Why gender and age prediction from tweets is hard: Lessons from a crowdsourcing experiment," in *Proc. 25th Int. Conf. Comput. Linguistics (COLING)*, 2014, pp. 1950–1961.
[17] E. Malmi and I. Weber, "You are what apps you use: Demographic prediction based on user's apps," in *Proc. Int. AAAI Conf. Web Social Media*, 2016, vol. 10, no. 1, pp. 635–638.
[18] S. C. Matz, J. I. Menges, D. J. Stillwell, and H. A. Schwartz, "Predicting individual-level income from Facebook profiles," *PLoS ONE*, vol. 14, no. 3, Mar. 2019, Art. no. e0214369.
[19] B. Kaur, D. Singh, and P. P. Roy, "Age and gender classification using brain–computer interface," *Neural Comput. Appl.*, vol. 31, no. 10, pp. 5887–5900, Mar. 2018.
[20] N. Bouadjenek, H. Nemmour, and Y. Chibani, "Fuzzy integrals for combining multiple SVM and histogram features for writer's gender prediction," *IET Biometrics*, vol. 6, no. 6, pp. 429–437, Nov. 2017.
[21] A. Jain and V. Kanhangad, "Gender classification in smartphones using gait information," *Expert Syst. Appl.*, vol. 93, pp. 257–266, Mar. 2018.
[22] V. Urbancokova, M. Kompan, Z. Trebulova, and M. Bielikova, "Behavior-based customer demography prediction in E-commerce," *J. Electron. Commerce Res.*, vol. 21, no. 2, pp. 96–112, 2020.
[23] A. Priadana, M. R. Maarif, and M. Habibi, "Gender prediction for Instagram user profiling using deep learning," in *Proc. Int. Conf. Decis. Aid Sci. Appl. (DASA)*, Nov. 2020, pp. 432–436.
[24] A. Pandya, M. Oussalah, P. Monachesi, and P. Kostakos, "On the use of distributed semantics of tweet metadata for user age prediction," *Future Gener. Comput. Syst.*, vol. 102, pp. 437–452, Jan. 2020.
[25] Y. Liu, H. Qu, W. Chen, and S. M. H. Mahmud, "An efficient deep learning model to infer user demographic information from ratings," *IEEE Access*, vol. 7, pp. 53125–53135, 2019.
[26] T. Van Hamme, G. Garofalo, E. Argones Rúa, D. Preuveneers, and W. Joosen, "A systematic comparison of age and gender prediction on IMU sensor-based gait traces," *Sensors*, vol. 19, no. 13, p. 2945, Jul. 2019.
[27] C. Suman, R. Chaudhari, S. Saha, S. Kumar, and P. Bhattacharyya, "Investigations in emotion aware multimodal gender prediction systems from social media data," *IEEE Trans. Computat. Social Syst.*, pp. 1–10, 2022.
[28] G. Antipov, S. A. Berrani, and J.-L. Dugelay, "Minimalistic CNN-based ensemble model for gender prediction from face images," *Pattern Recognit. Lett.*, vol. 70, pp. 59–65, Jan. 2016.
[29] P. Kaushik, A. Gupta, P. P. Roy, and D. P. Dogra, "EEG-based age and gender prediction using deep BLSTM-LSTM network model," *IEEE Sensors J.*, vol. 19, no. 7, pp. 2634–2641, Apr. 2019.
[30] J. H. Suh, "Machine-learning-based gender distribution prediction from anonymous news comments: The case of Korean news portal," *Sustainability*, vol. 14, no. 16, p. 9939, Aug. 2022.
[31] M. S. Qureshi, A. Aljarbouh, M. Fayaz, M. Bilal, W. Khan, and J. Khan, "An efficient methodology for water supply pipeline risk index prediction for avoiding accidental losses," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 5, pp. 1–9, 2020.
[32] B. Oh, J. Hwang, S. Seo, S. Chun, and K.-H. Lee, "Inductive Gaussian representation of user-specific information for personalized stress-level prediction," *Expert Syst. Appl.*, vol. 178, Sep. 2021, Art. no. 114912.
[33] R. G. Guimaraes, R. L. Rosa, D. De Gaetano, D. Z. Rodriguez, and G. Bressan, "Age groups classification in social network using deep learning," *IEEE Access*, vol. 5, pp. 10805–10816, 2017.
[34] Y. Dong, Y. Yang, J. Tang, Y. Yang, and N. V. Chawla, "Inferring user demographics and social strategies in mobile social networks," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2014, pp. 15–24.
[35] Y. Choi, Y. Kim, S. Kim, K. Park, and J. Park, "An on-device gender prediction method for mobile users using representative wordsets," *Expert Syst. Appl.*, vol. 64, pp. 423–433, Dec. 2016.
[36] E. Jahani, P. Sundsøy, J. Bjelland, L. Bengtsson, A. Pentland, and Y.-A. de Montjoye, "Erratum to: Improving official statistics in emerging markets using machine learning and mobile phone data," *EPJ Data Sci.*, vol. 6, no. 1, Jun. 2017.
[37] A. Ulges, D. Borth, and M. Koch, "Content analysis meets viewers: Linking concept detection with demographics on Youtube," *Int. J. Multimedia Inf. Retr.*, vol. 2, no. 2, pp. 145–157, Dec. 2012.
[38] I. M. Al-Zuabi, A. Jafar, and K. Aljoumaa, "Predicting customer's gender and age depending on mobile phone data," *J. Big Data*, vol. 6, no. 1, p. 18, Feb. 2019.
[39] J. Blumenstock, G. Cadamuro, and R. On, "Predicting poverty and wealth from mobile phone metadata," *Science*, vol. 350, no. 6264, pp. 1073–1076, Nov. 2015.
[40] R. Hirt, N. Kühl, and G. Satzger, "Cognitive computing for customer profiling: Meta classification for gender prediction," *Electron. Markets*, vol. 29, no. 1, pp. 93–106, Feb. 2019.
[41] A. Rosenfeld, S. Sina, D. Sarne, O. Avidov, and S. Kraus, "WhatsApp usage patterns and prediction of demographic characteristics without access to message content," *Demographic Res.*, vol. 39, pp. 647–670, Sep. 2018.
[42] A. Culotta, N. K. Ravi, and J. Cutler, "Predicting Twitter user demographics using distant supervision from website traffic data," *J. Artif. Intell. Res.*, vol. 55, pp. 389–408, Feb. 2016.
[43] M. M. Tadesse, H. Lin, B. Xu, and L. Yang, "Personality predictions based on user behavior on the Facebook social media platform," *IEEE Access*, vol. 6, pp. 61959–61969, 2018.
[44] Z. Wood-Doughty, N. Andrews, R. Marvin, and M. Dredze, "Predicting Twitter user demographics from names alone," in *Proc. 2nd Workshop Comput. Model. People's Opinions, Personality, Emotions Social Media*, 2018, pp. 105–111.
[45] A. Figueroa, B. Peralta, and O. Nicolis, "Coming to grips with age prediction on imbalanced multimodal community question answering data," *Information*, vol. 12, no. 2, p. 48, Jan. 2021.
[46] E. Ahmed, I. Yaqoob, and I. A. T. Hashem, "Recent advances and challenges in mobile big data," *IEEE Commun. Mag.*, vol. 56, no. 2, pp. 102–108, Feb. 2018.
[47] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Frechen, Germany: MITP, 2018.
[48] D. Butler, "When Google got flu wrong," *Nature*, vol. 494, no. 7436, pp. 155–156, Feb. 2013.
[49] T. W. Smith, "Survey-research paradigms old and new," *Int. J. Public Opinion Res.*, vol. 25, no. 2, pp. 218–229, Dec. 2012.
[50] T. P. Johnson and T. W. Smith, "Big data and survey research: Supplement or substitute?" in *Seeing Cities Through Big Data* (Springer Geography). Cham, Switzerland: Springer, Oct. 2016, pp. 113–125.
[51] H. Guo, B. Chen, R. Tang, W. Zhang, Z. Li, and X. He, "An embedding learning framework for numerical features in CTR prediction," in *Proc. 27th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2021, pp. 2910–2918.
[52] Y. Juan, Y. Zhuang, W.-S. Chin, and C.-J. Lin, "Field-aware factorization machines for CTR prediction," in *Proc. 10th ACM Conf. Recommender Syst.*, Sep. 2016, pp. 43–50.
[53] R. Kohavi and M. Sahami, "Error-based and entropy-based discretization of continuous features," in *Proc. KDD*, 1996, pp. 114–119.
[54] C. R. de Sá, C. Soares, and A. Knobbe, "Entropy-based discretization methods for ranking data," *Inf. Sci.*, vol. 329, pp. 921–936, Feb. 2016.
[55] J. Zhang, B. Bai, Y. Lin, J. Liang, K. Bai, and F. Wang, "General-purpose user embeddings based on mobile app usage," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 2831–2840.
[56] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.
[57] G. Zhou, X. Zhu, C. Song, Y. Fan, H. Zhu, X. Ma, Y. Yan, J. Jin, H. Li, and K. Gai, "Deep interest network for click-through rate prediction," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 1059–1068.
[58] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. 30th Int. Conf. Mach. Learn.*, 2013, vol. 30, no. 1, p. 3.
[59] Z. Tu, R. Li, Y. Li, G. Wang, D. Wu, P. Hui, and L. Su, "Your apps give you away: Distinguishing mobile users by their app usage fingerprints," in *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 2, no. 3, pp. 1–23, Sep. 2018.

[60] P. Welke, I. Andone, K. Blaszkiewicz, and A. Markowetz, "Differentiating smartphone users by app usage," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput.*, Sep. 2016, pp. 519–523.
[61] P. Unal, T. T. Temizel, and P. E. Eren, "What installed mobile applications tell about their owners and how they affect users' download behavior," *Telematics Inform.*, vol. 34, no. 7, pp. 1153–1165, Nov. 2017.
[62] J. Khan, M. Fayaz, A. Hussain, S. Khalid, W. K. Mashwani, and J. Gwak, "An improved alpha beta filter using a deep extreme learning machine," *IEEE Access*, vol. 9, pp. 61548–61564, 2021.
[63] X. He, S. Bowers, J. Q. Candela, J. Pan, O. Jin, T. Xu, B. Liu, T. Xu, Y. Shi, A. Atallah, and R. Herbrich, "Practical lessons from predicting clicks on ads at Facebook," in *Proc. 20th ACM SIGKDD Conf. Knowl. Discovery Data Mining (ADKDD)*, 2014, pp. 1–9.
[64] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006.
[65] C. Cortes and V. Vapnik, "Support vector machine," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995.
[66] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–9.
[67] O. Barkan and N. Koenigstein, "ITEM2 VEC: Neural item embedding for collaborative filtering," in *Proc. IEEE 26th Int. Workshop Mach. Learn. Signal Process. (MLSP)*, Sep. 2016, pp. 1–6.

XINGYU CHEN received the B.Sc. and M.Sc. degrees in electronic engineering from Tsinghua University, China, in 2018 and 2021, respectively. He is currently working as an AI Engineer with the Department of User and Market Research, China Mobile Research Institute. His research interests include mobile computing, recommendation systems, and deep learning.

YE GUO received the M.Sc. degree in signal and information processing from the Beijing University of Posts and Telecommunications, China, in 2010. She is currently working as a Chief Researcher with the Department of User and Market Research, China Mobile Research Institute. Her research interests include big data, data mining, and artificial intelligence.

HONGLEI XU received the M.Sc. degree in library information and archive management from the Beijing Institute of Technology, China, in 2017. She is currently working as an AI Engineer with the Department of User and Market Research, China Mobile Research Institute. Her research interests include mobile computing, natural language processing, and machine learning.



HONGYAN YAN received the M.Sc. degree from the University of International Business and Economics, China. She is currently working as a Chief Expert with the Department of User and Market Research, China Mobile Research Institute. Her research interests include user research and big data.

LIN LIN received the M.Sc. degree in communication from the Communication University of China, China, in 2002. She is currently working as the Vice Director with the Department of User and Market Research, China Mobile Research Institute. Her research interests include big data, marketing research, and digital strategy.