Multimodal LLM-Based Agent for Human Behavior Simulation: Modeling Return Migration Dynamics

Xiaoluan Liu[1†], Xinyu Lin[2], Fangbin Qiao[1]

[1] *Central University of Finance and Economics, Beijing 100081, China;*
[2] *National University of Singapore, Singapore 119077*

Abstract

Accurately modeling and simulating complex human mobility is pivotal for evidence-based socioeconomic planning, yet remains under-explored in the era of Large Language Models (LLMs). We introduce the Return Migration Simulation (RMS) task, which focuses on predicting individual decisions to move from urban back to rural regions—a process critical for understanding urban–rural dynamics and formulating balanced development policies. The key to the RMS task lies in the in-depth reasoning over multimodal features to capture human intention and predict the individual decision. To this end, we present RMS-Agent, an LLM-powered agent endowed with latent reasoning capability. RMS-Agent first encodes multimodal features through the heterogeneous data tokenizer, where we specifically design a tabular tokenizer to convert structured table features into dense vectors compatible with the LLM. To achieve comprehensive and in-depth reasoning, we propose using multiple meta-queries to probe the LLM to reason and uncover latent intention and predict migration decision. Extensive experiments on three real-world datasets demonstrate that RMS-Agent significantly outperforms competitive machine-learning and deep-learning baselines across accuracy, F1, and AUC metrics, verifying its capacity to capture nuanced migration drivers. To summarize, this work (i) formulates a novel return migration simulation task, (ii) proposes a generalizable LLM-based agent architecture for multimodal latent reasoning, and (iii) provides a comprehensive benchmark with substantial empirical exploration for this socially significant problem, laying the groundwork for richer human-mobility modeling with LLMs in the future.

Keywords: LLM-based Agent; Multimodal Data Modeling; Heterogeneous Data Tokenization; Human Behavior Simulation; Latent Reasoning

1. Introduction

Understanding the mechanisms behind complex human behaviors and forecasting future trajectories represent fundamental challenges in computational social science. Human behavior simulation with neural models has emerged as a transformative paradigm to address the challenges [1, 2]. Crucially, the fidelity of human behavior simulation depends on the model's capacity to capture the fundamental mechanisms driving such behavior. As such, through high-fidelity

[†]Corresponding author: Xiaoluan Liu (Email: xiaoluanliu@163.com; OCRID: 0009-0003-2499-2091)

behavioral simulation, researchers can uncover latent behavioral patterns, infer causal relationships, and predict future behavioral dynamics. Existing human behavior simulations span a broad spectrum of applications, including but not limited to modeling social interactions [3, 4], consumption behaviors [5, 6], and urban mobility patterns [7]. These efforts provide valuable insights for social and economic sciences, offering significant implications for understanding complex societal systems and formulating public policies.

Technically speaking, existing human behavior simulation approaches have evolved through three significant stages:

- Machine Learning-based Behavior Simulation: Early work primarily employs machine learning techniques such as Naïve Bayes classifiers and shallow neural networks for behavior prediction in several constrained domains like web search and browsing behavior. While computationally efficient, these approaches exhibit limited simulation performance due to their inability to capture complex behavioral patterns.

- Deep Learning-based Behavior Simulation: The advent of deep learning introduces more powerful models, for example, Recurrent Neural Network variants (*e.g.,* GRU [8], LSTM [9]), Transformer [10], and pre-training techniques [11]. These enable superior behavioral representation learning across broader simulation scenarios such as consumption patterns [12], social interactions [3], and urban mobility [7]. Nevertheless, previous deep learning models such as LSTM [13] and BERT [11] still lack rich world knowledge and strong reasoning and generalization capabilities.

- LLM-powered Agent for Behavior Simulation: With the emergence of Large Language Models (LLMs), modern simulation agents demonstrate unprecedented capabilities in contextual understanding, deliberative reasoning, and independent interactions with environments. For instance, Park *et al.* (2023) [14] and Wang *et al.* (2023) [15] demonstrated that LLM-based agents can simulate realistic human behaviors and social phenomena through memory-enhanced planning and controllable sandbox environments. At the multi-agent level, frameworks like CAMEL [16] and AgentSociety [17] demonstrate how LLM-powered agents can simulate complex social dynamics and collective behaviors.

However, existing studies on LLM-powered agents for human behavior simulation have predominantly focused on simulating social media interactions, urban transportation, conversation, or consumption behaviors. They notably neglect the essential simulation of *human mobility behaviors*, especially the migration between urban and rural areas. Spatial migration behaviors constitute the physical substrate that affects all location-dependent activities, such as social interactions, urban transportation, and commercial behaviors. More importantly, simulating human migration is indispensable for elucidating fundamental migration mechanisms, informing evidence-based socioeconomic policy design, and optimizing urban-rural infrastructure [18]. In recent years, some developing nations (particularly China) have increasingly focused on return migration, the reverse movement of the population from urban to rural areas, investigating its underlying drivers, and then implementing targeted policies to promote coordinated urban-rural development [19].

To bridge this research gap, we propose the task of utilizing LLM-powered agents to simulate human return migration behaviors. Specifically, as shown in Figure 1, this task predicts individual return migration decisions (urban → rural) through multimodal reasoning across heterogeneous features, including tabular demographic attributes, textual semantic descriptions (*e.g.,*

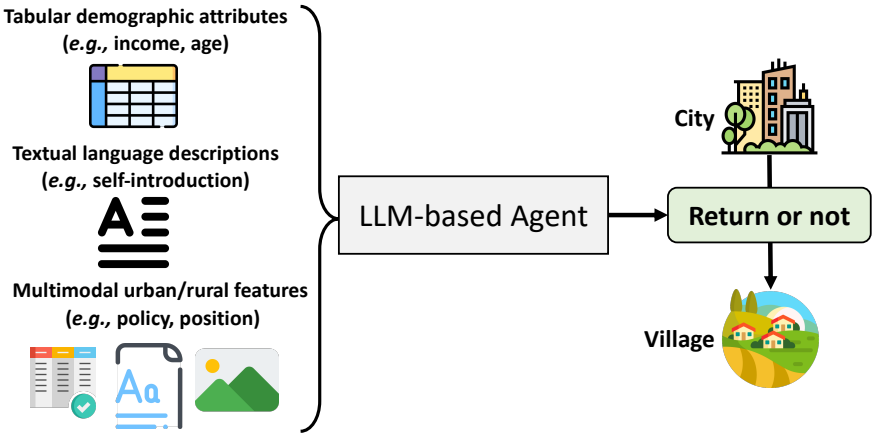

Figure 1: Illustration of the task of using the LLM-based agent for return migration simulation. It integrates the multimodal features of individual, family, urban area, and rural area to predict the human's return migration decision from urban to rural areas.

profile or policy statements), and multimodal urban/rural characteristics. The key to this task is to reason over these heterogeneous features from individual, family, and regional perspectives to infer latent human intentions and decision-making policies. However, directly applying explicit Chain-of-Thought (CoT) reasoning of LLMs to this task presents significant challenges: the absence of verifiable human feedback prevents reliable supervision of the reasoning process for uncovering genuine migration intentions.

To address the challenge, we propose RMS-Agent: an LLM-based agent with latent reasoning ability for Return Migration Simulation (RMS). The RMS-Agent first encodes multimodal features through heterogeneous data tokenizers and then performs latent reasoning via the LLM to deeply infer heterogeneous signals, thereby predicting individual intentions and migration decisions. In particular, we design a tabular tokenizer through a Multi-Layer Perception (MLP) to translate non-semantic tabular data (*e.g.,* anonymous IDs and categorical features) or other multimodal input (*e.g.,* image) into dense vectors for LLM understanding. Thereafter, multiple meta-query vectors are fed into an LLM for in-depth multi-step reasoning for intention discovery and return migration prediction. We conduct extensive experiments on three real-world datasets, demonstrating the significant effectiveness of our proposed RMS-Agent. We release our code and data to facilitate future research.

To sum up, the contributions of this work are threefold:

- We first propose the task of utilizing LLM-based agents for return migration simulation, highlighting the significance and challenges of modeling return migration dynamics.
- We propose RMS-Agent, an LLM-based agent to execute latent reasoning over the multimodal and heterogeneous features for return migration simulation.
- We conduct extensive experiments on three datasets under various settings, validating the superior performance over traditional machine learning and deep learning baselines.

2. Related Work

In this section, we review three key stages in the development of human behavior modeling and simulation: early approaches based on traditional machine learning, subsequent advances driven by deep learning models, and recent progress in human behavior simulation enabled by LLMs and agents.

2.1. Machine Learning for Behavior Simulation

In recent years, with the rapid development of machine learning techniques, behavior prediction has become an increasingly important direction in user modeling. By learning patterns from historical behavioral data, machine learning models can effectively anticipate users' future actions. This approach captures evolving user interests and behavioral trends and has been widely applied in personalized recommendation, search optimization, human-computer interaction, and user behavior simulation.

With advances in artificial intelligence and machine learning, academic efforts in behavior modeling have gradually shifted from heuristic rules to data-driven approaches. In the early stage of user modeling, Jennings and Higuchi (1993) pioneered the use of neural networks in building personalized news services, demonstrating the potential of neural models to continuously learn user preferences [20]. Lieberman (1995) developed the Letizia system, a web browsing assistant that proactively recommended hyperlinks by tracking and predicting user interests, marking an important step toward intelligent and personalized web agents [21]. In 1997, Pazzani and Billsus proposed a user modeling algorithm based on the Naïve Bayes classifier, which enabled incremental learning from user feedback and became a widely adopted technique due to its efficiency [22]. In 1998, Davison and Hirsh introduced an algorithm for predicting sequences of user actions, demonstrating strong performance on large UNIX command datasets [23]. That same year, Horvitz *et al.* proposed the Lumiere Project, which applied Bayesian networks to infer user intentions, serving as the foundation for Microsoft Office 97's Office Assistant and representing a key milestone in the commercialization of behavior modeling [24]. The year 2001 marked an important stage in the theoretical formalization of this field. Webb *et al.* systematically established a machine learning framework for user modeling, emphasizing the need for dynamic model updates to accommodate changes in user behavior [25]. In parallel, Zukerman and Albrecht proposed a predictive statistical modeling paradigm, offering a probabilistic foundation for user modeling and strengthening the theoretical underpinnings of statistical approaches [26]. Together, these contributions laid a systematic methodological framework for behavior prediction. As research evolved, focus shifted from explicit modeling to the inference of latent behavior in complex environments. In 2002, Bonabeau introduced agent-based modeling (ABM), which emphasized individual-level interactions and system-level dynamics, opening new directions for modeling complex social behaviors [27]. In 2005, Shen *et al.* proposed a user modeling method based on implicit feedback, analyzing click behavior and contextual signals [28]. Their UCAIR client-side search agent demonstrated significant improvements in personalized search performance. In 2008, Ziebart *et al.* introduced a framework based on maximum entropy inverse optimal control, framing user behavior as a context-sensitive decision-making process [29]. By learning conditional probabilistic models, their approach enhance both the accuracy and interpretability of human behavior predictions and expanded the theoretical boundaries of behavior modeling. In 2018, Rabinowitz *et al.* introduced the concept of "Machine Theory of Mind," proposing the ToMnet neural architecture that enables machines to infer the mental states of other agents based on observed behavior [30]. This work markes a shift in

user modeling from surface-level behavioral prediction toward modeling cognitive mechanisms, initiating a new phase in artificial intelligence that emphasizes understanding human intentions and contextual adaptation. In subsequent developments, Abri *et al.* (2020) systematically categorized user modeling methods for personalized web search, analyzing the characteristics and application scenarios of various techniques and providing a comprehensive reference for future research [31].

Overall, user behavior modeling and prediction have progressed from early methods based on Naïve Bayes and shallow neural networks to more sophisticated techniques such as sequence modeling and Bayesian inference. The field has gradually embraced agent-based simulations, implicit feedback learning, and cognitive modeling to improve the explanatory power and generalizability of behavior models. While these advancements have led to significant progress, limitations remain. Early models such as Naïve Bayes and shallow neural networks struggle to capture complex nonlinear relationships and contextual dependencies. Moreover, traditional models often rely on static features, making them inadequate for modeling the dynamic evolution of user preferences over time. As Abri *et al.* (2020) pointed out, early user modeling methods tend to overlook the sequential and temporal nature of behavior, limiting their real-world applicability [31].

2.2. Deep Learning for Behavior Simulation

In response to the limitations of traditional machine learning in capturing complex behavioral patterns, deep learning-based behavior modeling has emerged. Deep neural networks, with their strong nonlinear modeling capabilities, have shown remarkable performance in behavior sequence prediction, cross-domain modeling, and capturing the evolution of user interests. In particular, recurrent neural networks (RNNs) such as LSTM and GRU, along with attention mechanisms and Transformer architectures, have significantly enhanced the expressiveness and adaptability of behavior models, leading to improvements in both accuracy and generalization [32, 12].

In recommendation systems, Elkahky *et al.* (2015) proposed a multi-view deep learning framework for cross-domain user modeling, improving recommendations for cold-start users [33]. Zhou *et al.* (2018) introduced the ATRank model with attention mechanisms to identify the most relevant segments of a user's historical behavior, significantly boosting recommendation accuracy [32]. That same year, the Deep Interest Network model (DIN) was proposed to capture the diversity and dynamics of user interests, demonstrating industrial applicability in click-through rate prediction [34]. Gu *et al.* (2020) proposed the Hierarchical User Profile (HUP) model using a pyramid-style RNN to capture multi-granular interest evolution [12]. Guo *et al.* (2018) extended modeling boundaries by integrating text and image information into a unified multimodal framework [35]. In social media and security contexts, Agarwal *et al.* (2022) and Toshevska *et al.* (2023) applied graph neural networks to detect spam behavior and antisocial content [3, 4], respectively. Meanwhile, variants of RNNs have been widely used to model the temporal dynamics of behavior sequences. Zhu *et al.* (2017) proposed Time-LSTM to capture both short- and long-term preferences using temporal intervals [13]. Building on this, Ren *et al.* (2019) developed a lifelong sequential modeling approach with a hierarchical memory network for personalized long-term behavior modeling [36]. More recently, Transformer-based architectures have become mainstream in user modeling. Qi *et al.* (2022) proposed the FUM model for news recommendation, using Fastformer to balance modeling accuracy with computational efficiency [37]. Wu *et al.* (2021) introduced TRISAN, a tri-relational spatiotemporal attention network that incorporates location information into behavior modeling for location-based search tasks [7].

Despite these advances, deep learning models for behavior prediction face several persistent challenges. First, current models are often task-specific (*e.g.,* CTR prediction or product recommendation), limiting their generalization and transferability across domains [38]. Second, their black-box nature and lack of interpretability hinder their adoption in sensitive decision-making scenarios [39].

To address these limitations, recent research has explored the application of LLMs in behavior modeling and prediction. With strong capabilities in knowledge representation, transfer learning, and generalizable reasoning, LLMs offer a promising alternative to traditional approaches. Unlike rule-based or shallow learning systems, LLM-powered simulation agents exhibit sophisticated contextual understanding, reasoning, and interactive abilities. A growing body of work has demonstrated the potential of LLMs to simulate cognitive, emotional, and behavioral patterns, enabling a paradigm shift from "scripted agents" to "human-like agents".

2.3. LLMs for Behavior Simulation

Research on LLMs for behavior simulation can be divided into three main categories. The first focuses on dialog simulation. For example, Mysore *et al.* (2023) introduced the LACE model with editable user profiles for controllable and interpretable text recommendation [40]. Zhang and Balog (2020) designed a framework to simulate user-system dialog interactions, offering a standardized method for evaluating conversational systems [41]. PlatoLM [42] used simulated users to train LLMs, achieving improved multi-turn dialog modeling. The second category explores social behavior simulation. Xie *et al.* (2024) examined whether GPT-4 agents can replicate human trust behavior and found a high level of behavioral alignment [43]. Park *et al.* (2023) proposed the "Generative Agents" architecture, equipping agents with memory, reflection, and planning to reproduce complex social interactions [14]. Piao *et al.* (2025) developed the "AgentSociety" platform with over 10,000 LLM agents to study social phenomena such as polarization and misinformation [17]. Gao *et al.* (2023) used the S³ system to simulate emotion contagion and gender bias in social networks [44]. Li *et al.* (2023) employed the CAMEL framework for multi-agent cooperation [16]. Park *et al.* (2022) and Mou *et al.* (2024) provided systematic classifications of individual, scene-based, and society-level modeling [45, 46]. Aher *et al.* (2023) and Argyle *et al.* (2023) used LLMs to replicate classical psychology and political science experiments, highlighting their potential as proxies for human participants in the social sciences [5, 6]. The third category focuses on economic and policy simulation. Horton (2023) proposed the concept of "homo silicus", showing that LLMs can reproduce many behavioral biases in economic decision-making [47]. Chu *et al.* (2023) trained LLMs on curated "media diets" to predict public opinion, offering a novel modeling tool for social science research [48].

Overall, these studies have addressed key bottlenecks in traditional behavior modeling, such as the lack of interpretability in user profiles, the inability to reconstruct group behavior, and the difficulty of linking individual actions to broader social dynamics. By serving as high-capacity, cognitively capable agents, LLMs enable user modeling to progress from shallow behavior fitting to deep cognitive simulation. Despite these advances, the current use of LLMs in behavior modeling has largely centered around domains such as social communication, dialog interaction, consumption patterns, and urban mobility. However, spatial migration—particularly movements between urban and rural areas—remains relatively underexplored in this line of research. As a core component of human spatial behavior, migration underpins a wide range of location-dependent activities and directly impacts regional planning, infrastructure allocation, and social policy design. In particular, return migration—where individuals move from cities back to rural

areas—has gained increasing attention in both policy and academic circles due to its implications for balanced regional development.

To extend the scope of LLM-driven behavior modeling, this paper introduces a novel simulation task focused on return migration. We explore how LLM-powered agents can be applied to capture the dynamics and motivations behind this form of spatial mobility, aiming to support the understanding of migration behavior and inform data-driven decision-making in population and urban-rural planning contexts.

3. Problem Formulation

In this section, we present the task formulation of utilizing LLM-based agents for return migration simulation. Formally, given an individual u in an urban area, we formulate the return migration simulation as a decision prediction task using an LLM-based agent. The goal is to predict whether u will migrate back to their rural origin, based on multimodal contextual features. The input multimodal contextual features include, but are not limited to:

- **Individual traits**: Demographic and economic attributes (*e.g.,* age, income, education).
- **Family factors**: Household composition and ties (*e.g.,* number of dependents, family occupation, household wealth).
- **Urban context**: City-specific push factors (*e.g.,* living cost, job satisfaction).
- **Rural context**: Village-specific pull factors (*e.g.,* economic opportunities, policy incentives).

The heterogeneous features mentioned above mainly cover tabular modality $\mathcal{X}_{tabular}$ and textual modality $\mathcal{X}_{text}$. We leave more data modalities to future exploration[1]. $\mathcal{X}_{tabular}$ includes some table data with anonymous IDs and categorical features, while $\mathcal{X}_{text}$ covers some semantic features and descriptions. Formally, the LLM-based agent first needs to tokenize heterogeneous inputs into the text representation space of LLMs for reasoning and prediction:

$$y_u = \text{LLM}_\theta(\text{Tokenize}(\mathcal{X}_{tabular}, \mathcal{X}_{text})), \tag{1}$$

where the LLM performs deep reasoning over tokenized multimodal features to discover human intention and predict the migration decision $y_u \in [0, 1]$. The ground-truth label of y_u is

$$y = \begin{cases} 1 & \text{(Return to rural area)}, \\ 0 & \text{(Stay in urban area)}. \end{cases} \tag{2}$$

4. Method

In this section, we detail how the proposed RMS-Agent harnesses multimodal tabular data ($\mathcal{X}_{\text{tabular}}$) and textual data ($\mathcal{X}_{\text{text}}$) to perform deep reasoning over human intention and predict the return migration behavior. Specifically, we introduce the tokenization of heterogeneous multimodal data in Section 4.1, followed by Section 4.2, which presents how RMS-Agent leverages meta-queries for latent reasoning over return migration intention.

[1]Although RMS-Agent is capable of tokenizing additional modalities such as images, we leave this for future work due to the absence of such data in existing datasets.

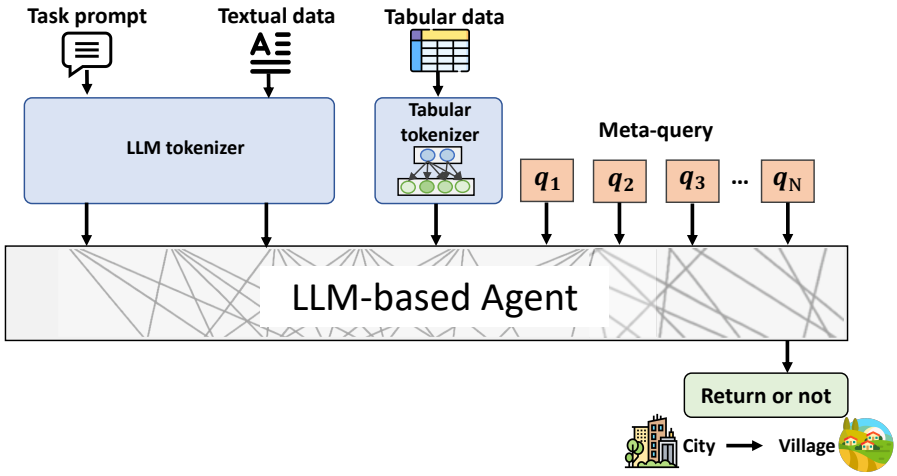

Figure 2: Illustration of RMS-Agent, which integrates the task prompt, textural data, and tabular data for in-depth reasoning with meta-query vectors to achieve return migration simulation.

4.1. Heterogeneous Data Tokenization

As shown in Figure 2, given the tabular data $\mathcal{X}_{\text{tabular}}$ and textual data $\mathcal{X}_{\text{text}}$, which contain individual, family, urban, and rural features, we employ two separate tokenizers to encode the heterogeneous inputs. For the textual modality $\mathcal{X}_{\text{text}}$, we use the tokenizer of the LLM itself (*e.g.,* the SentencePiece for Qwen3 [49]) to align with the pre-trained semantic space, thereby enabling better utilization of its rich world knowledge. Formally, we concatenate the task prompt p with $\mathcal{X}_{\text{text}}$, and feed it into the LLM tokenizer to obtain a token vector sequence:

$$\{\mathbf{t}_1, \ldots, \mathbf{t}_M\} \leftarrow \text{LLM\_Tokenizer}([p; \mathcal{X}_{\text{text}}]), \tag{3}$$

where $\mathcal{X}_{\text{text}}$ covers the textual data such as self-introduction and rural/urban policy. Besides, we implement the task prompt p using the following template to illustrate the task of return migration simulation:

For tabular data $\mathcal{X}_{\text{tabular}}$, many features are represented as anonymized codes and categorical features (*e.g.,* regions or years), which are often not semantically meaningful in the LLM's pre-training space. Directly applying the LLM's tokenizer may result in these codes being split into subwords with irrelevant meanings, leading to potential information loss. To address this, we design a separate neural network to learn a dedicated tabular tokenizer. Specifically, we convert the tabular features of each individual through one-hot encoding and organize them into a single column vector, and then feed it into the tabular tokenizer to produce a dense token vector $\mathbf{v}$:

$$\mathbf{v} \leftarrow \text{Tabular\_Tokenizer}(\mathcal{X}_{\text{tabular}}),$$

where the tabular tokenizer is instantiated by an MLP in this work, and $\mathbf{v}$ encodes the tabular information with the same dimensionality as the token vectors in $\{\mathbf{t}_1, \ldots, \mathbf{t}_M\}$.

• **Task prompt:** "You are a sociologist studying rural labor migration. Your task is to determine, based on an individual's basic information, whether this person is a "returnee"—that is, whether they will return to live or work in their original rural hometown after working in the urban area.
Please decide whether this individual is a "returnee" based on the given features. Your answer must be strictly "Yes" or "No". Do not include any explanations, reasons, or additional words.

Below is the individual's profile: Gender: values["gender"];
Age: values["age"];
Years of Education: values["education"];
Household Registration Type: values["hukou"];
Marital Status: values["marriage"];
Physical Condition: values["physical_condition"];
Work Experience: values["work"];
After-Tax Monthly Wage (RMB): values["wage"] ;
Type of Pension Insurance: values["old_age_insurance"];
Type of Medical Insurance: values["medical_insurance"];
Hospitalization Expenses Last Year (RMB): values["hospitalization_expenses"];
Unemployment Insurance: values["unemployment_insurance"];
Housing Provident Fund: values["housing_fund"];
Household Size: values["hh_size"];
From a Low-Income Household: values["poor_hh"];
Housing Type: values["house_type"];
Number of Cars Owned: values["count_car"];
Phone Type: values["phone_type"];
Total Household Assets (RMB): values["total_asset"];
Total Household Income (RMB): values["total_income"];
Total Household Debt (RMB): values["total_debt"];
Total Household Consumption (RMB): values["total_consumption"];
...{other semantic features}

Based on the above information, is this person a "returnee"? Please answer only with "Yes" or "No"."

*4.2. **Latent Reasoning***

Given the tokenized sequences $\{\mathbf{t}_1, \ldots, \mathbf{t}_M\}$ and $\mathbf{v}$, we feed them into the LLM and leverage meta-queries to guide the LLM in performing in-depth reasoning from multiple perspectives within the latent space; thereafter, the LLM makes the final prediction for the return migration behavior. Formally, we have

$$\mathbf{h} \leftarrow \mathrm{LLM}_\theta(\mathbf{t}_1, \ldots, \mathbf{t}_M, \mathbf{v}, \mathbf{q}_1, \ldots, \mathbf{q}_N), \tag{4}$$

where $\mathbf{h}$ denotes the hidden state of the last layer of the LLM corresponding to the last meta-query token. $\{\mathbf{q}_1, \ldots, \mathbf{q}_N\}$ is the sequence of N meta-queries, which are randomly initialized learnable vectors. Such meta-queries are initialized with randomness to maintain diversity, which encourages the LLM to reason over the inputs $\{\mathbf{t}_1, \ldots, \mathbf{t}_M\}$ and $\mathbf{v}$ from multiple perspectives through the self-attention in Transformer [10]. As illustrated in Figure 3, different meta-queries

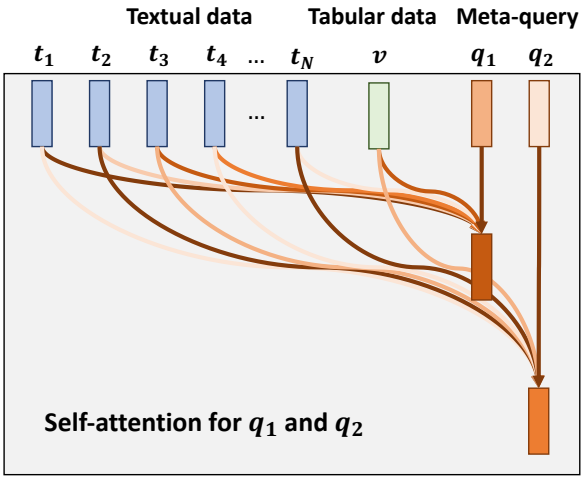

Figure 3: An example of attention weight visualization for meta-queries $\mathbf{q}_1$ and $\mathbf{q}_2$ only. Note that the attention weights for other vectors such as $\mathbf{t}_1$ and $\mathbf{v}$ are omitted for simplicity. Due to the different initialization, $\mathbf{q}_1$ and $\mathbf{q}_2$ can attend to different features for in-depth reasoning in the latent space.

attend to distinct subsets of features, enabling the LLM to perform comprehensive and in-depth reasoning. Finally, $\mathbf{h}$ encodes the reasoning results of the LLM to predict the return migration behavior:

$$y_u = \sigma(\text{Linear}(\mathbf{h})), \tag{5}$$

where $\text{Linear}(\cdot)$ represents a linear projection layer with learnable parameters $\mathbf{W}$ and $\mathbf{b}$; and y_u denotes the predicted return migration behavior. Besides, $\sigma(\cdot)$ is the Sigmoid function to restrict the prediction within $[0, 1]$.

Training. Given the training data $\{(\mathcal{X}^i_{\text{text}}, \mathcal{X}^i_{\text{tabular}}, y_i)\}_{i=1}^I$, we optimize the following parameters of RMS-Agent: meta-queries $\{\mathbf{q}_1, \ldots, \mathbf{q}_N\}$, the LLM's θ, $\mathbf{W}$, and $\mathbf{b}$. We use the cross-entropy loss function as follows:

$$-\frac{1}{I}\sum_{i=1}^{I}\left[y_i \log(y_{u,i}) + (1 - y_i)\log(1 - y_{u,i})\right], \tag{6}$$

where $y_{u,i}$ is the prediction of RMS-Agent for the feature input $\mathcal{X}^i_{\text{text}}$ and $\mathcal{X}^i_{\text{tabular}}$; and y_i is the ground truth behavior label.

5. Experiment

5.1. Experimental Settings

- **Datasets.** We utilize data from the China Household Finance Survey (CHFS), a nationally representative household-level survey administered by the Survey and Research Center for China Household Finance at Southwestern University of Finance and Economics [2]. Specifically, we use

[2]https://chfser.swufe.edu.cn/datas/Products/Datas/DataList/.

Table 1: Datasets Statistics.

Dataset	# Samples	# Positive Samples	# Negative Samples
CHFS-2015	18,396	7,201	11,195
CHFS-2017	23,225	10,295	12,930
CHFS-2019	21,182	9,190	11,992

three survey waves: 2015, 2017, and 2019. For each wave, we extract a set of variables harmonized across years to ensure consistency in our analysis, containing rich information including but not limited to the following categories:

- **Individual traits:** gender, age (restricted to 16–60), years of education, marital status, physical condition, household size, and phone type.
- **Family factors:** poverty status, house ownership type, number of cars owned, total household assets, income, debt, and consumption.
- **Urban and Rural context:** employment status, wage, type of old-age pension, medical insurance, unemployment insurance, and housing fund participation.

To ensure robustness and prevent potential information leakage, we perform several data cleaning steps. First, we remove three post-return variables (*i.e.,* the number of years since return, the province lived in before return, and the job type before return). Second, we exclude individuals who have never migrated, as our study focuses on return migration behavior. Lastly, we discard noisy samples that are simultaneously marked as both returnee and migrant, which indicates logical inconsistencies. After preprocessing, we randomly split the samples into training, validation, and testing sets with a ratio of 8:1:1. The final cleaned datasets used for experiments and analysis are summarized in Table 1.

- **Baselines**. We compare RMS-Agent with competitive baselines, including traditional machine learning methods (Logistic Regression, Random Forest, Gradient Boosting, XGBoost, SVM), deep learning-based methods (MLP, BERT), and LLM-based methods (Prompt, SFT).

Traditional machine learning methods. 1) **Logistic Regression** [50] is a widely used linear model for binary classification. 2) **Random Forest** [51] is an ensemble of decision trees that captures non-linear feature interactions through bagging. 3) **Gradient Boosting** [52] improves prediction by sequentially correcting errors of weak learners. 4) **XGBoost** [53] is an optimized gradient boosting framework with regularization. 5) **SVM** (Support Vector Machine) [54] is a margin-based classifier effective in high-dimensional feature spaces.

Deep learning-based methods. 6) **MLP** [55] is a multi-layer perceptron that encodes tabular features into deep latent representations for classification. 7) **BERT** [11] is a pre-trained language model adapted to this task by finetuning on input prompts that combine textual and structured features, enabling the model to leverage contextual understanding.

LLM-based methods. 8) **Prompt** reformulates the classification task as language modeling and performs zero-shot inference using a pre-trained LLM. 9) **Supervised Fine-Tuning (SFT)** further trains the LLM on labeled decision data using task-specific prompts, enabling the model to specialize for the migration prediction task. In this work, we utilize Qwen3 [49] as the backbone to implement the above two methods.

- **Evaluation Metrics.** We adopt five widely used metrics to evaluate the models, including 1) *Accuracy* measures the proportion of correctly predicted instances among all samples; 2) *Precision* indicates the proportion of true positives among all predicted positives; 3)*Recall* reflects the proportion of true positives that are correctly identified among all actual positives; 4)*F1* Score is the harmonic mean of precision and recall, balancing both metrics; 5) *AUC* evaluates the model's ability to distinguish between classes across different thresholds.

- **Implementation Details.** For ***traditional machine learning baselines***, we adopt standard implementations from `scikit-learn`[3]. Logistic regression uses L2 regularization with the default solver. Random Forest and Gradient Boosting are configured with 100 estimators, and the learning rate for XGBoost is fixed at 0.1. All input features are preprocessed via median imputation and standardization for numerical features, and most-frequent imputation with one-hot encoding for categorical features. For ***deep learning baselines***, we employ a two-layer MLP with hidden sizes of 126 and 64, trained using Adam with a fixed learning rate of 1×10^{-3}. For the transformer-based BERT model, the input embedding dimension is 128, the batch size is 64, and the encoder uses multi-head self-attention with 2 heads. Parameters are initialized using Xavier and Kaiming schemes. For ***RMS-Agent***, we use Qwen2.5-1.5B-Insturct[4] as the backbone LLM and adopt parameter-efficient tuning method LoRA [56] to fine-tune the model. For the tabular input, we use a one-layer MLP with 128 hidden size followed by the hidden dimension of the backbone LLM. The learning rate is set at 1×10^{-3}, and we tune the number of meta-query vectors N in $\{2, 3, 4, 5, 6, 8\}$. The best hyper-parameters are selected based on validation performance.

5.2. Overall Performance

The overall performance comparison between baselines and our proposed RMS-Agent is presented in Table 2, from which we have the following observations:

- Traditional machine learning methods, including logistic regression, random forests, and boosting-based models, demonstrate moderate performance across all datasets. Among the traditional machine learning models, ensemble-based methods (*i.e.,* Random Forest and XGBoost), generally outperform simpler linear models, reflecting their capacity to capture non-linear interactions among structured features. However, their improvements remain limited, suggesting challenges in modeling more complex behavioral patterns inherent in return migration decisions. This aligns with expectations given the structured nature of the input but also highlights the limitations of relying solely on static decision boundaries.

- Deep learning baselines, particularly MLP and BERT, exhibit stronger performance than traditional methods in most cases, especially on CHFS-2015 and CHFS-2019. MLP benefits from its capacity to learn higher-order feature interactions, while BERT shows potential in incorporating semantic context despite being primarily designed for textual data. Nevertheless, their results are not consistently superior across all datasets, and both models occasionally underperform in recall or AUC, indicating sensitivity to dataset characteristics and possible underutilization of contextual signals.

[3] `https://scikit-learn.org/stable/`.
[4] `https://huggingface.co/Qwen/Qwen2.5-1.5B-Instruct`.

Table 2: Overall performance comparison between baselines and our proposed RMS-Agent. The best results are in bold, and the second-best results are underlined. The results highlight the superiority of our proposed RMS-Agent over these baselines across three datasets.

Dataset	Method	Accuracy	Precision	Recall	F1 Score	AUC
CHFS-2015	**Logistic Regression**	0.7859	0.7754	0.6428	0.7029	0.8627
	Random Forest	0.8092	0.7766	0.7241	0.7495	0.8872
	Grandient Boosting	0.7891	0.7573	0.6841	0.7188	0.8717
	XGBoost	0.7962	0.7660	0.6952	0.7289	0.8775
	SVM	0.7951	0.7458	0.7283	0.7369	0.8661
	MLP	0.8125	0.7827	0.7255	0.7530	0.8878
	BERT	0.8033	0.7227	0.8124	0.7649	0.8697
	Prompt	0.6429	0.6349	0.2207	0.3275	0.7823
	SFT	0.7163	0.5818	**0.9959**	0.7345	0.7792
	RMS-Agent	**0.8625**	**0.8164**	0.8400	**0.8280**	**0.9331**
CHFS-2017	**Logistic Regression**	0.7090	0.7065	0.8273	0.7621	0.7606
	Random Forest	0.7305	0.7207	0.8518	0.7808	0.8038
	Grandient Boosting	0.7107	0.7094	0.8243	0.7625	0.7724
	XGBoost	0.7193	0.7206	0.8197	0.7670	0.7733
	SVM	0.7150	0.7161	0.8189	0.7641	0.7915
	MLP	0.7335	**0.7796**	0.7349	0.7566	0.8189
	BERT	0.7223	0.7283	0.8090	0.7666	0.7867
	Prompt	0.5562	0.5826	0.7487	0.6553	0.5349
	SFT	0.5635	0.6485	0.4920	0.5595	0.5739
	RMS-Agent	**0.7628**	0.7445	**0.8816**	**0.8073**	**0.8286**
CHFS-2019	**Logistic Regression**	0.7664	0.7333	0.7130	0.7230	0.8478
	Random Forest	0.7768	0.7253	0.7693	0.7467	0.8601
	Grandient Boosting	0.7683	0.6895	0.8333	0.7546	0.8529
	XGBoost	0.7688	0.6894	**0.8355**	0.7555	0.8551
	SVM	0.7735	0.6976	0.8300	0.7581	0.8509
	MLP	0.8023	0.7560	0.7936	0.7744	0.8758
	BERT	0.6535	0.7218	0.6264	0.6708	0.6988
	Prompt	0.5819	0.5085	0.6611	0.5749	0.6262
	SFT	0.7249	0.6681	0.7086	0.6877	0.7228
	RMS-Agent	**0.8150**	**0.7672**	0.8146	**0.7901**	**0.8816**

- Our proposed RMS-Agent consistently achieves the best overall performance across all datasets and evaluation metrics. The improvements are particularly notable in terms of balanced classification metrics such as F1 and AUC, underscoring the model's ability to integrate multimodal features and perform context-aware reasoning. These results support the hypothesis that LLM-based agents are better equipped to simulate human decisions in complex social contexts, as they can capture subtle dependencies and latent factors beyond surface-level correlations.

- In addition to achieving strong overall performance, RMS-Agent maintains a favorable balance across precision, recall, and AUC. Unlike methods that exhibit extreme biases, such as high recall at the cost of precision, RMS-Agent delivers stable and well-rounded results, which is critical in decision-making tasks where both false positives and false negatives carry significant implications. This consistency across metrics and datasets demonstrates the model's robustness and suitability for real-world return migration simulation.

- Beyond the general trends across model families, we observe that performance differences are more pronounced on CHFS-2015 and CHFS-2019 than on CHFS-2017. This may reflect

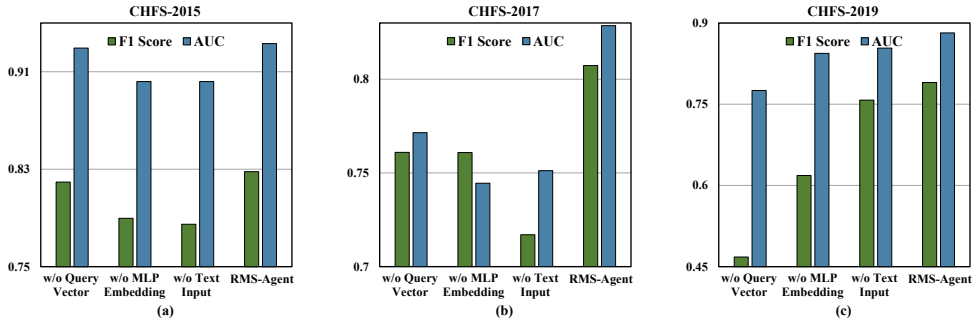

Figure 4: Ablation study on three datasets. The results validate the effectiveness of using meta-queries, MLP encoding, and textual data in RMS-Agent.

temporal or data quality variations, such as richer contextual features or clearer migration patterns in certain years.

- We evaluate prompt-based inference and SFT to assess the adaptability of vanilla LLMs. While SFT improves over prompting in recall and F1, both methods underperform compared to traditional and deep learning baselines. This highlights the limitations of using generic LLMs without structured alignment or reasoning. The results confirm the necessity of our design in RMS-Agent, which tightly integrates multimodal tokenization with task-specific reasoning.

5.3. In-depth Analysis

5.3.1. Ablation Study

To assess the effectiveness of each component in RMS-Agent, we conduct an ablation study on the three datasets. Specifically, we consider three ablated variants of our model: 1) w/o Query Vector, which removes the task-specific meta-query vector used for latent reasoning; 2) w/o MLP Embedding, which excludes the structured tabular input by omitting its MLP-based encoding; and 3) w/o Text Input, which removes the textual context provided to the LLM. These variants are compared against the full RMS-Agent model to examine the relative importance of each component. We evaluate performance using the representative F1 score and AUC to capture both classification accuracy and ranking capability.

From the results reported in Figure 4, we can find that 1) each component plays a critical role in the overall effectiveness of RMS-Agent. Removing the meta-query vector leads to a consistent drop in both F1 and AUC across all datasets, highlighting its importance for guiding the model's reasoning process. 2) The absence of MLP embedding, which removes structured tabular data, results in a more significant performance degradation—particularly in F1—indicating that individual- and household-level features are crucial for accurate return migration prediction. 3) Notably, removing textual input causes the most substantial decline, especially on CHFS-2017 and CHFS-2019, suggesting that rural-urban contextual descriptions provide key information that complements structured attributes. These patterns reinforce the importance of multimodal integration and justify our design of a unified architecture that combines tabular inputs, textual context, and query-guided reasoning.

5.3.2. Token Generation versus Disrciminative Prediction

To study the effectiveness of the discriminative binary classifier to align with the task objective, we compare the settings of 1) generative head and 2) discriminative head. Precisely, for

		Accuracy	Precision	Recall	F1 Score	AUC
CHFS-2015	**Discriminative**	0.8141	0.7391	0.8166	0.7759	0.8932
	RMS-Agent	**0.8625**	**0.8164**	**0.8400**	**0.8280**	**0.9331**
CHFS-2017	**Discriminative**	0.6642	0.6393	0.9274	0.7569	0.6259
	RMS-Agent	**0.7628**	**0.7445**	**0.8816**	**0.8073**	**0.8286**
CHFS-2019	**Discriminative**	0.7381	0.6618	0.7925	0.7212	0.7450
	RMS-Agent	**0.8150**	**0.7672**	**0.8146**	**0.7901**	**0.8816**

Table 3: Performance comparison between token generation (discriminative) and discriminative prediction (RMS-Agent) on three datasets.

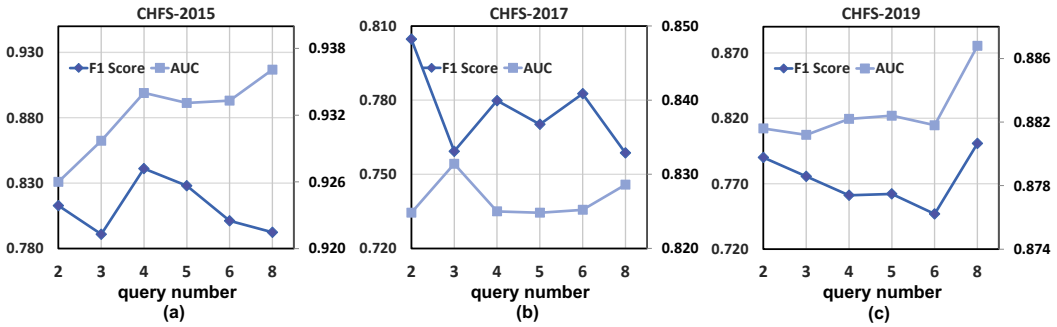

Figure 5: Effect of the meta-query number.

the generative head setting, we utilize the original LLM generation head to generate the token of "yes" and "no". The LLM is optimized via log likelihood maximization. For the discriminative head setting, we utilize a binary classifier in our RMS-Agent, optimized by the cross-entropy loss as in Eq. (6). The goal is to assess whether explicitly modeling binary decisions improves alignment between reasoning and prediction, which is particularly relevant for structured tasks like return migration.

From the results, we can observe that 1) the discriminative head consistently outperforms the generative one across all datasets, especially in F1 and AUC. This suggests that direct supervision helps the model focus on decision boundaries, while the generative approach may dilute learning signals due to vocabulary-wide token prediction. Additionally, 2) the generative head tends to favor high recall but suffers from low precision, indicating less controlled decision calibration. Therefore, we empirically hypothesize that task-specific heads are crucial for classification tasks with reasoning components.

5.3.3. *Effect of Meta-query Number*

To study the effect of query number, we evaluate our model with different numbers of query vectors from 2 to 8 on three datasets. The experimental results are shown in Figure 5. We can find that: 1) increasing the number of queries generally leads to improved AUC, particularly on CHFS-2015 and CHFS-2019. This suggests that using more query vectors enhances the model's ability to perform in-depth reasoning. 2) However, the F1 score does not always increase with more queries. In fact, performance often peaks at moderate values (*e.g.,* 4 or 5 query vectors), suggesting that too many queries may introduce redundancy or noise. This trade-off highlights the importance of selecting an appropriate number of queries for balanced classification performance. Overall, we empirically find that the number of queries is a critical factor that affects model performance, and tuning it appropriately can significantly improve both discrimination

Table 4: Performance comparison between RMS-Agent using Qwen2.5 of different model sizes.

Model Size	Accuracy	Precision	Recall	F1 Score	AUC
1.5B	0.8625	0.8164	0.8400	0.8280	0.9331
3B	0.8560	0.7662	0.9131	0.8332	0.9334
7B	0.8625	0.8057	0.8579	0.8310	0.9315

Table 5: Generalization ability of RMS-Agent on CHFS-2017 and CHFS-2019. "RMS-Agent-2015", "RMS-Agent-2017", and "RMS-Agent-2019" denote RMS-Agent trained on 2015, 2017, and 2019, respectively.

		Accuracy	Precision	Recall	F1 Score	AUC
CHFS-2017	**Random Forest**	0.6403	0.6782	0.7198	0.682	0.6753
	SVM	0.6375	0.6788	0.7013	0.6732	0.6691
	RMS-Agent-2017	**0.7628**	**0.7445**	**0.8816**	**0.8073**	**0.8286**
	RMS-Agent-2015	0.6625	0.692	0.7227	0.707	0.7058
	Relative Decline	13.15%	7.05%	18.02%	12.42%	14.82%
		Accuracy	**Precision**	**Recall**	**F1 Score**	**AUC**
CHFS-2019	**Random Forest**	0.677	0.6406	0.5854	0.6259	0.7681
	MLP	0.663	0.6412	0.5801	0.627	0.7422
	RMS-Agent-2019	**0.815**	**0.7672**	**0.8146**	**0.7901**	**0.8816**
	RMS-Agent-2015	0.7225	0.6963	0.6225	0.6573	0.7841
	Relative Decline	11.35%	9.24%	23.58%	34.27%	11.06%

(AUC) and calibration (F1) metrics.

*5.3.4. **Effect of LLM Size***

To investigate whether larger model sizes bring stronger reasoning capability, we compare RMS-Agent based on Qwen2.5 models with 1.5B, 3B, and 7B parameters. From the results in Table 4, we observe that 1) scaling up from 1.5B to 3B leads to performance gains in F1 score and AUC, suggesting that larger models benefit from richer pre-training data and greater expressiveness, enabling stronger reasoning over user features. 2) However, further scaling to 7B does not yield consistent improvements. This is possibly due to the use of LoRA, which may limit fine-tuning capacity to bridge the task gap between next-token generation and binary classification. This indicates that scaling model size alone is not sufficient; more data or superior alignment methods are needed for superior performance scaling in the future.

*5.3.5. **Generalization Ability of RMS-Agent***

To study the generalization ability of RMS-Agent across different distributions, we trained the RMS-Agent on 2015 data and evaluate it on the CHFS-2017 and CHFS-2019. We denote RMS-Agent trained on CHFS- 2015, 2017, and 2019 data as "RMS-Agent-2015", "RMS-Agent-2017", and "RMS-Agent-2019", respectively. The results are shown in Table 5. From the results, we can observe that: 1) RMS-Agent trained on the 2015 data demonstrates stronger generalization ability compared to other baselines. This can be attributed to the rich world knowledge encoded in LLMs, which enables the model to capture more robust decision features rather than relying on superficial correlations. Nevertheless, 2) RMS-Agent-2015 struggles to compete with RMS-Agent-2017 and RMS-Agent-2019 on the 2017, and 2019 data, respectively. This is reasonable since there might be a temporal data distribution shift from 2015 to 2019. The relative performance decline of RMS-Agent-2015 compared with RMS-Agent-2017 (on data 2017) and RMS-Agent-2019 (on data 2019) partially explains this, *i.e.,* the relative decline on 2017 data is generally smaller than that on 2019.

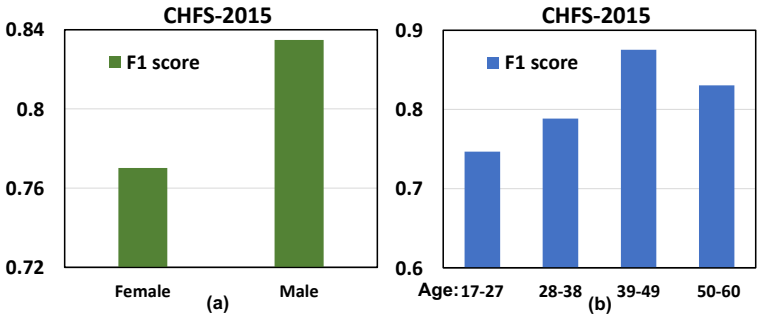

Figure 6: Performance comparison across different age and gender groups.

Table 6: Performance comparison between MLP, RMS-Agent, and RMS-Agent using additional temporal modality data (RMS-Agent-T) on CHFS-2015 dataset.

Models	Accuracy	Precision	Recall	F1 Score	AUC
MLP	0.8125	0.7827	0.7255	0.7530	0.8878
RMS-Agent	0.8625	**0.8164**	0.8400	0.8280	0.9331
RMS-Agent-T	**0.8631**	0.8079	**0.8428**	**0.8316**	**0.9365**

5.3.6. *User Group Analysis*

To further analyze the influential features that drive the return migration, we evaluate RMS-Agent on different user groups, according to gender and age, respectively. From the results in Figure 6, we can observe that male migrants, middle-aged individuals (39–49), and elderly individuals (50–60) show higher model performance. This could be because the return migration behaviors of these demographic groups are more predictable, partly validating the necessity of incorporating these demographic features.

5.3.7. *Effectiveness of Multimodal Data*

To study whether additional multimodal data help RMS-Agent achieve better simulation, we incorporate temporal data as an additional modality into our RMS-Agent. Specifically, we use the individual living trajectory, which contains provinces each individual have stayed in, in a chronological order. From the results shown in Table 6, we can find that incorporating temporal data further enhances the performance of RMS-Agent in most cases, which verifies the effectiveness of other modalities for return migration simulation. This further strengthens our statements on the effectiveness of using meta-queries to capture different subsets of multimodal features and extract the influential information for prediction.

6. Conclusion and Future Work

To simulate return migration behaviors, this work presents RMS-Agent, a novel LLM-powered agent with latent reasoning capability over heterogeneous and multimodal features. RMS-Agent integrates multimodal data, such as tabular and textual data, via the heterogeneous data tokenizer. Besides, it employs multiple meta-queries to perform in-depth reasoning and uncover latent migration intention. Extensive experiments on real-world datasets demonstrate substantial performance gains over existing baselines, highlighting the potential of LLMs in high-fidelity human behavior simulation. This study lays a foundation for leveraging LLMs to model complex socio-spatial phenomena and simulate human mobility behaviors due to LLMs' richer contextual understanding and generalization capabilities.

Moving forward, we aim to extend RMS-Agent in several directions. First, incorporating richer multimodal data such as temporal dynamics and longitudinal mobility data could enhance the agent's capacity to model evolving intention. Second, integrating explicit human feedback or preference signals in the right way might improve the interpretability and controllability of the reasoning of the RMS-Agent. Third, leveraging the agent's retrieval tools to gather additional information about urban and rural regions from the real-world websites could further enrich the simulation context. Fourth, we plan to develop additional benchmark datasets on human physical mobility behaviors to support future research, for instance, rural-to-urban migration, inter-city migration, and rural-to-rural migration. We can also explore the effectiveness of RMS-Agent across these diverse tasks.

Code and Data Availability

To facilitate reproduction, we release our code and data at `https://github.com/Linxyhaha/RMS-Agent`.

Author Contributions

X.L. Liu is responsible for Conceptualization, Methodology, Experiment, and Writing. X.Y. Lin is for Data curation, Experiment, and Writing; F.B. Qiao works on Investigation, Supervision, Writing-review, and Editing.

Acknowledgements

We would like to express our gratitude to the editors and reviewers for taking the time to review our paper.

References

[1] Pedro Lara-Benítez, Manuel Carranza-García, and José C Riquelme. An experimental review on deep learning architectures for time series forecasting. *International journal of neural systems*, 31(03):2130001, 2021.

[2] Massimiliano Luca, Gianni Barlacchi, Bruno Lepri, and Luca Pappalardo. A survey on deep learning for human mobility. *ACM Computing Surveys (CSUR)*, 55(1):1–44, 2021.

[3] Prabhat Agarwal, Manisha Srivastava, Vishwakarma Singh, and Charles Rosenberg. Modeling user behavior with interaction networks for spam detection. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, pages 2437–2442, 2022.

[4] Martina Toshevska, Slobodan Kalajdziski, and Sonja Gievska. Graph neural networks for antisocial behavior detection on twitter. In *International Conference on ICT Innovations*, pages 222–236. Springer, 2023.

[5] Gati V Aher, Rosa I Arriaga, and Adam Tauman Kalai. Using large language models to simulate multiple humans and replicate human subject studies. In *International Conference on Machine Learning*, pages 337–371. PMLR, 2023.

[6] Lisa P Argyle, Ethan C Busby, Nancy Fulda, Joshua R Gubler, Christopher Rytting, and David Wingate. Out of one, many: Using language models to simulate human samples. *Political Analysis*, 31(3):337–351, 2023.

[7] Yi Qi, Ke Hu, Bo Zhang, Jia Cheng, and Jun Lei. Trilateral spatiotemporal attention network for user behavior modeling in location-based search. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 3373–3377, 2021.

[8] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[9] Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. A review of recurrent neural networks: Lstm cells and network architectures. *Neural computation*, 31(7):1235–1270, 2019.

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, volume 30, 2017.

[11] Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the conference of the North American chapter of the association for computational linguistics*, pages 4171–4186, 2019.

[12] Yulong Gu, Zhuoye Ding, Shuaiqiang Wang, and Dawei Yin. Hierarchical user profiling for e-commerce recommender systems. In *Proceedings of the 13th international conference on web search and data mining*, pages 223–231, 2020.

[13] Yu Zhu, Hao Li, Yikang Liao, Beidou Wang, Ziyu Guan, Haifeng Liu, and Deng Cai. What to do next: Modeling user behaviors by time-lstm. In *IJCAI*, volume 17, pages 3602–3608. Melbourne, VIC, 2017.

[14] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, pages 1–22, 2023.

[15] Lei Wang, Jingsen Zhang, Hao Yang, Zhiyuan Chen, Jiakai Tang, Zeyu Zhang, Xu Chen, Yankai Lin, Ruihua Song, Wayne Xin Zhao, et al. User behavior simulation with large language model based agents. *arXiv preprint arXiv:2306.02552*, 2023.

[16] Guohao Li, Hasan Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. Camel: Communicative agents for" mind" exploration of large language model society. *Advances in Neural Information Processing Systems*, 36:51991–52008, 2023.

[17] Jinghua Piao, Yuwei Yan, Jun Zhang, Nian Li, Junbo Yan, Xiaochong Lan, Zhihong Lu, Zhiheng Zheng, Jing Yi Wang, Di Zhou, Chen Gao, Fengli Xu, Fang Zhang, Ke Rong, Jun Su, and Yong Li. Agentsociety: Large-scale simulation of llm-driven generative agents advances understanding of human behaviors and society, 2025.

[18] Guy J Abel, Michael Brottrager, Jesus Crespo Cuaresma, and Raya Muttarak. Climate, conflict and forced migration. *Global environmental change*, 54:239–249, 2019.

[19] Sylvie Démurger and Hui Xu. Return migrants: The rise of new entrepreneurs in rural china. *World Development*, 39(10):1847–1861, 2011.

[20] Andrew Jennings and Hideyuki Higuchi. A user model neural network for a personal news service. *User Modeling and User-Adapted Interaction*, 3:1–25, 1993.

[21] Henry Lieberman et al. Letizia: An agent that assists web browsing. *IJCAI (1)*, 1995:924–929, 1995.

[22] Michael Pazzani and Daniel Billsus. Learning and revising user profiles: The identification of interesting web sites. *Machine learning*, 27:313–331, 1997.

[23] Brian D Davison and Haym Hirsh. Predicting sequences of user actions. In *Notes of the AAAI/ICML 1998 workshop on predicting the future: AI approaches to time-series analysis*, pages 5–12, 1998.

[24] Eric J Horvitz, John S Breese, David Heckerman, David Hovel, and Koos Rommelse. The lumiere project: Bayesian user modeling for inferring the goals and needs of software users. *arXiv preprint arXiv:1301.7385*, 2013.

[25] Christos Papatheodorou. Machine learning in user modeling. pages 286–294, 2001.

[26] George Karypis and Mukund Deshpande. A feature-based approach to recommending selections based on text understanding. *International Journal of Adaptive and Intelligent Systems*, 11(1–2):5–18, 2001.

[27] Eric Bonabeau. Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the national academy of sciences*, 99(suppl_3):7280–7287, 2002.

[28] Xuehua Shen, Bin Tan, and ChengXiang Zhai. Implicit user modeling for personalized search. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM05, pages 824–831. ACM, October 2005.

[29] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. Human behavior modeling with maximum entropy inverse optimal control. In *AAAI spring symposium: human behavior modeling*, volume 92, 2009.

[30] Neil Rabinowitz, Frank Perbet, Francis Song, Chiyuan Zhang, SM Ali Eslami, and Matthew Botvinick. Machine theory of mind. In *International conference on machine learning*, pages 4218–4227. PMLR, 2018.

[31] Sara Abri, Rayan Abri, and Salih Çetin. A classification on different aspects of user modelling in personalized web search. In *Proceedings of the 4th International Conference on Natural Language Processing and Information Retrieval*, pages 194–199, 2020.

[32] Chang Zhou, Jinze Bai, Junshuai Song, Xiaofei Liu, Zhengchao Zhao, Xiusi Chen, and Jun Gao. Atrank: An attention-based user behavior modeling framework for recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[33] Ali Mamdouh Elkahky, Yang Song, and Xiaodong He. A multi-view deep learning approach for cross domain user modeling in recommendation systems. In *Proceedings of the 24th international conference on world wide web*, pages 278–288, 2015.

[34] Guorui Zhou, Na Mou, Ying Fan, Qi Pi, Weijie Bian, Chang Zhou, Xiaoqiang Zhu, and Kun Gai. Deep interest evo-

lution network for click-through rate prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 5941–5948, 2019.

[35] Yangyang Guo, Zhiyong Cheng, Liqiang Nie, Xin-Shun Xu, and Mohan Kankanhalli. Multi-modal preference modeling for product search. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 1865–1873, 2018.

[36] Kan Ren, Jiarui Qin, Yuchen Fang, Weinan Zhang, Lei Zheng, Weijie Bian, Guorui Zhou, Jian Xu, Yong Yu, Xiaoqiang Zhu, et al. Lifelong sequential modeling with personalized memorization for user response prediction. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 565–574, 2019.

[37] Tao Qi, Fangzhao Wu, Chuhan Wu, and Yongfeng Huang. Fum: fine-grained and fast user modeling for news recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, pages 1974–1978, 2022.

[38] Jie Gu, Feng Wang, Qinghui Sun, Zhiquan Ye, Xiaoxiao Xu, Jingmin Chen, and Jun Zhang. Exploiting behavioral consistence for universal user representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 4063–4071, 2021.

[39] Erasmo Purificato, Ludovico Boratto, and Ernesto William De Luca. Do graph neural networks build fair user models? assessing disparate impact and mistreatment in behavioural user profiling. In *Proceedings of the 31st ACM international conference on information & knowledge management*, pages 4399–4403, 2022.

[40] Sheshera Mysore, Mahmood Jasim, Andrew McCallum, and Hamed Zamani. Editable user profiles for controllable text recommendations. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 993–1003, 2023.

[41] Shuo Zhang and Krisztian Balog. Evaluating conversational recommender systems via user simulation. In *Proceedings of the 26th acm sigkdd international conference on knowledge discovery & data mining*, pages 1512–1520, 2020.

[42] Chuyi Kong, Yaxin Fan, Xiang Wan, Feng Jiang, and Benyou Wang. Platolm: Teaching llms in multi-round dialogue via a user simulator. *arXiv preprint arXiv:2308.11534*, 2023.

[43] Chengxing Xie, Canyu Chen, Feiran Jia, Ziyu Ye, Shiyang Lai, Kai Shu, Jindong Gu, Adel Bibi, Ziniu Hu, David Jurgens, et al. Can large language model agents simulate human trust behavior? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[44] Chen Gao, Xiaochong Lan, Nian Li, Yuan Yuan, Jingtao Ding, Zhilun Zhou, Fengli Xu, and Yong Li. Large language models empowered agent-based modeling and simulation: A survey and perspectives. *Humanities and Social Sciences Communications*, 11(1):1–24, 2024.

[45] Joon Sung Park, Lindsay Popowski, Carrie Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Social simulacra: Creating populated prototypes for social computing systems. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, pages 1–18, 2022.

[46] Xinyi Mou, Xuanwen Ding, Qi He, Liang Wang, Jingcong Liang, Xinnong Zhang, Libo Sun, Jiayu Lin, Jie Zhou, Xuanjing Huang, et al. From individual to society: A survey on social simulation driven by large language model-based agents. *arXiv preprint arXiv:2412.03563*, 2024.

[47] John J Horton. Large language models as simulated economic agents: What can we learn from homo silicus? Technical report, National Bureau of Economic Research, 2023.

[48] Eric Chu, Jacob Andreas, Stephen Ansolabehere, and Deb Roy. Language models trained on media diets can predict public opinion. *arXiv preprint arXiv:2303.16779*, 2023.

[49] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[50] Sandro Sperandei. Understanding logistic regression analysis. *Biochemia medica*, 24(1):12–18, 2014.

[51] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[52] Alexey Natekin and Alois Knoll. Gradient boosting machines, a tutorial. *Frontiers in neurorobotics*, 7:21, 2013.

[53] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[54] Mayank Arya Chandra and SS Bedi. Survey on svm and their application in image classification. *International Journal of Information Technology*, 13(5):1–11, 2021.

[55] Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in neural information processing systems*, 34:24261–24272, 2021.

[56] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Author Biography

Xiaoluan Liu is a Ph.D. candidate at the Central University of Finance and Economics, under the supervision of Prof. Fangbin Qiao. Her research interests lie in labor mobility, agricultural economic development, labor economics, and machine learning.

Xinyu Lin is a Ph.D. candidate at the National University of Singapore, under the supervision of Prof. Tat-Seng Chua. Her research interests lie in LLM-based social modeling, and her work has been published in top conferences and journals such as SIGIR, WWW, and TOIS. Moreover, she has served as the reviewer and PC member for the top conferences such as SIGIR, WWW, and KDD.

Fangbin Qiao is a professor at Central University of Finance and Economics. His research interests lie in development economics, agricultural economics, environmental and resource economics, econometrics, etc. So far, he has more than 50 papers published at high-level journals.