A Novel Approach of Reputation Trust Metric Based on Knowledge Graph Combined with Large Language Model for Improving Trust Worthiness Calculation on Social Networks

Duc Hieu Nguyen Huy, Trong Mai Phu, Quyen Nguyen Thuong
hieunhdhe182545@fpt.edu.vn, FPT University, Ha Noi

Abstract

In today's rapidly evolving world, social media platforms are integral to our daily lives, serving as gateways to a variety of services and transactions. Yet, the anonymity allowed on these platforms poses significant security risks, despite the presence of preventive measures. The vast amount of data and the continual evolution of anonymity tactics present considerable challenges in managing these risks. Consequently, there is a growing interest in developing methods to assess the trustworthiness of interactions within social networks. This study introduces a method that leverages large language models, combined with reputation-based trust systems and knowledge graphs, to evaluate the reliability of services on social media. The framework is structured into three modules: data collection, data processing, and trust computation model development. The effectiveness of the proposed model is demonstrated through an experimental study, utilizing a dataset derived from service providers on the Facebook social network.

Key words: Trustworthiness Model, Social Network, Large Language Model, Knowledge Graph, Sentiment Analysis, Reputation Trust

I. INTRODUCTION

In recent times, the explosive development of information technology and the internet has created many new opportunities for people. Social networks, a product of this progress, have become an indispensable part of everyday life. It not only helps connect people from all over the world but also provides a platform for information sharing, entertainment, and even business. With social networks, people can easily interact, learn and expand their relationship network quickly and effectively. Social networks are no longer simply a “virtual world”, all activities on these platforms are now like real-world interactions, directly impacting daily life. Although there are many advantages, it is due to the continuous exchanges and interactions Human customs in social networks but also more problems arise, one of the most talked about concepts in today's social network environment is reliability.

In the context of social networks, trust is described as a measure of the degree of confidence that members or organizations will behave in an expected way. Trust is considered crucial to the success of social networks because it affects users' willingness to share and interact within these platforms. Trustworthiness on social networks significantly affects business campaigns, identifying online fraud, etc. Because of such direct impacts, calculating the trustworthiness of participating individuals is difficult. Social networks are extremely important. In social media, trust is understood as the willingness of an individual (trustor) to accept risk based on subjective trust in another person (trustee), with the expectation that this person will act reliably, optimizing benefits to the trustee in uncertain situations. Measuring and estimating reliability requires quantifying it through specific models and formulas, which can vary according to specific circumstances and applications. There are many applications used to evaluate the credibility of information on social networks, including tools such as spam account detection, fake news identification, retweet analysis, recommendation systems, and dispute resolution. spread of false information.

In this research article, we will mention more clearly the concept of reliability: reputation trust is a concept related to the trust that people place in an individual, organizations or brands based on their reputation. This reputation is built over time through their actions, interactions, and consistent behavior on social media. The article focuses on evaluating the reputation of traders in a group on the social network Facebook by using information about that trader's past interactions, thereby determining the level of reputation of traders in the group. Here we have focused on developing and applying large language models (solving problems related to semantics), prompting techniques and using knowledge graphs to calculate the reliability of information. each merchant on social networks from the data we collect.

To summarize, our research has the following main contributions:

- We applied LLM (Large Language Models) to identify the contexts of interactions of social network posts and comments.
- Modeling social network data into knowledge graphs, preprocessing to remove duplicate data, seeding posts and comments.
- Building a TFTModel model to calculate reputation trust score by combining two aspects service trust and interaction trust based on improved PageRank and determining the corresponding interaction weight.

The remainder of the article will be organized as follows. The next part of the article will be related works. In part III, we will explain the method of utilizing prompt engineering with large language model in data preprocessing and information extraction. In Part IV we will focus on explaining the process and how we calculate reputation trust score. Part V will evaluate the results of the algorithm with other methods. And the last part is our conclusion for this model.

II. RELATED WORKS

Calculating reputation trust score involves quantifying the concept of trust and the entities involved in it. In this context, when we say A trusts B, we understand that A is the trustor and B is the trustee. The trustor evaluates the trustworthiness between themselves and the trustee based on their own knowledge and experience with the trustee, and possibly based on the opinions of others about the trustee. In this scenario (where trust exists between two individuals), one party is defined as the "trustor" (the one assessing trustworthiness) and the other as the "trustee" (the one being assessed for trustworthiness) [6].

Social media data is highly diverse, complex, and heterogeneous. Each social network like Facebook, Twitter, Epinion, and Amazon has its unique characteristics, so calculating trustworthiness heavily depends on the data used for simulation. This study focuses on the diversity and complexity of data in assessing trustworthiness, primarily through social trust and service quality [22]. In the process of calculating trustworthiness, data related to interactions and relationships between the Trustor and Trustee can be affected by temporal and contextual factors [6]. Trust relationships can change over time and social contexts, as well as specific situations that impact trust assessments, including influences from external individuals and circumstances that create trust [7]. Some algorithms used previously in calculating reputation trust score will be mentioned in more detail in the **Appendix A Section: Some related models in trust calculation.**

III. PROMPT ENGINEERING IN TRUST COMPUTATION MODEL

A. The effect of using LLM in trust computation

There are not as many options available as in the trustworthiness calculation models. Validating connections is a challenging task. Online social networks include many types of connections among individuals and groups. Besides explicit connections like emotions and ratings, interactions containing semantic content are difficult to determine the extent of influence. Although many studies have used techniques such as natural language processing (NLP) to determine the level of influence. These methods are effective in certain limited linguistic contexts. The effects of information exchange on online social networks are incredibly diverse. They require classification models to be trained with large and varied datasets. This is often difficult to achieve using traditional classification methods employing deep learning. Recently, large language models (LLM) have emerged as a powerful tool in semantic analysis applications on heterogeneous data. The Large Language Model (LLM) is an artificial intelligence model that has made significant strides in understanding and processing natural language, making large language models a versatile and powerful tool. The LLM is built based on a deep neural network architecture like the Transformer, then trained with a massive amount of data from websites, newspapers, documents, etc. [35] Famous current large language models include GPT-3.5, GPT-4, Claude AI, etc. Trained with vast available datasets, LLMs not only understand language deeply but also perform various tasks from answering questions to content creation, translation, and summarizing texts [36]. Due to this superiority, LLMs open new doors to automation and improving tasks related to language and text processing [37].

When using LLMs for specific problems, LLMs have several advantages over conventional deep learning models:

- LLMs do not require retraining with large datasets to solve specific problems.
- LLMs can handle diverse data types that are not homogeneous.
- LLMs can be quickly fine-tuned for specific tasks.
- LLMs are capable of handling a wide range of tasks, including complex ones.

B. LLM Model

In this study, we use the Vistral-7B-chat model [56], a multi-way conversational large language model for Vietnamese. For more details about this model, you can see in **Appendix B Section: Vistral-7B-Chat Vietnamese LLM Model.**

C. Apply Prompt Engineering in Trust computation

The most apparent application of LLMs today is in semantic analysis [36]. Trained with large datasets, LLMs can deeply analyze conversations, understand the context thoroughly, and analyze complex text structures. However, using LLMs for context analysis involves significant reliance on providing actions (Prompts) for the LLM, and incorrect actions can lead to results that deviate from the desired outcomes [38]. Therefore, controlling LLMs with actions is extremely important, as incorrect actions can reduce the accuracy of LLMs and affect the accuracy of the problem being solved. We can see that Large language models (LLMs) perform tasks entirely depending on the actions provided by the user. Incorrect actions will lead to skewed results, while carefully considered actions will yield more accurate results [38]. Because of the importance of providing actions, the field of Prompt Engineering has emerged. Prompt Engineering is an engineering field that uses language effectively to manipulate AI models, with the ability to articulate in writing being a central skill. To create clear and effective prompts, engineers must understand the nuances of language and express their intentions accurately [39]. There are many techniques used to improve accuracy in prompt engineering, but in this proposal, we will discuss two main methods: Few-shot Prompting [50] and Prompt Chaining [51]. More details about these two types of prompt techniques will be presented in the **Appendix C section: Prompt Engineering Techniques.**

In our research, the LLM model and prompt engineering techniques to quantify text data with indicators can be included in the model for calculation. The LLM model we use in two main parts of the problem is **Service Exploration** and **Feedback Evaluation** as shown in **Fig. 1.** and **Fig. 2**. During the **Service Exploration** process, we use the input as the content of the posts, then these posts will be passed through an LLM model combined with the use of prompt techniques such as Few-Shot Prompting and Prompting chain to show service providers and the types of services they provide.

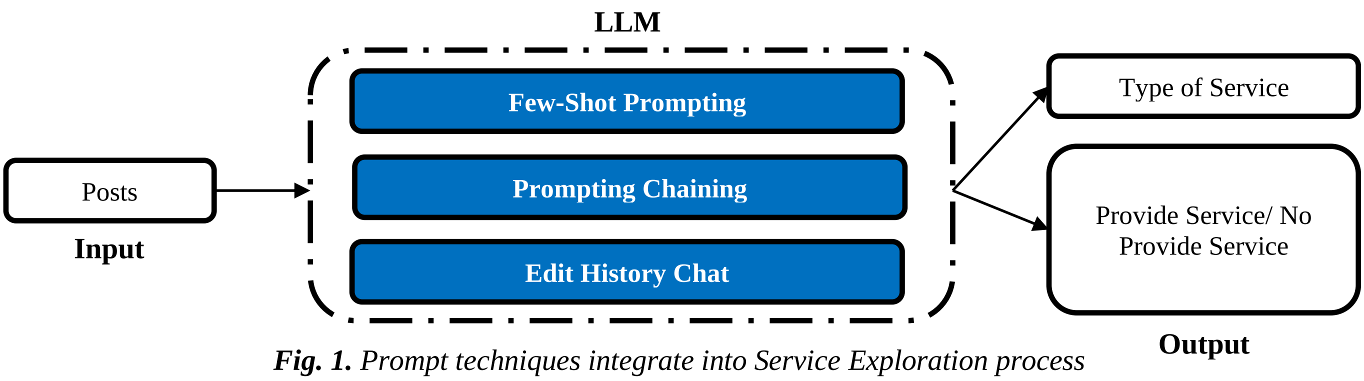

Fig. 1. *Prompt techniques integrate into Service Exploration process*

As for the **Feedback Evaluation** module, the input here is defined as comments, then these comments are also passed through the LLM model combined with the Few-Shot Prompting technique to give evaluations about that comment, if comments numbered -5 means negative comments, numbered 0 means non-related comments, numbered 5 means positive comments, numbered 10 are comments asking about the service.

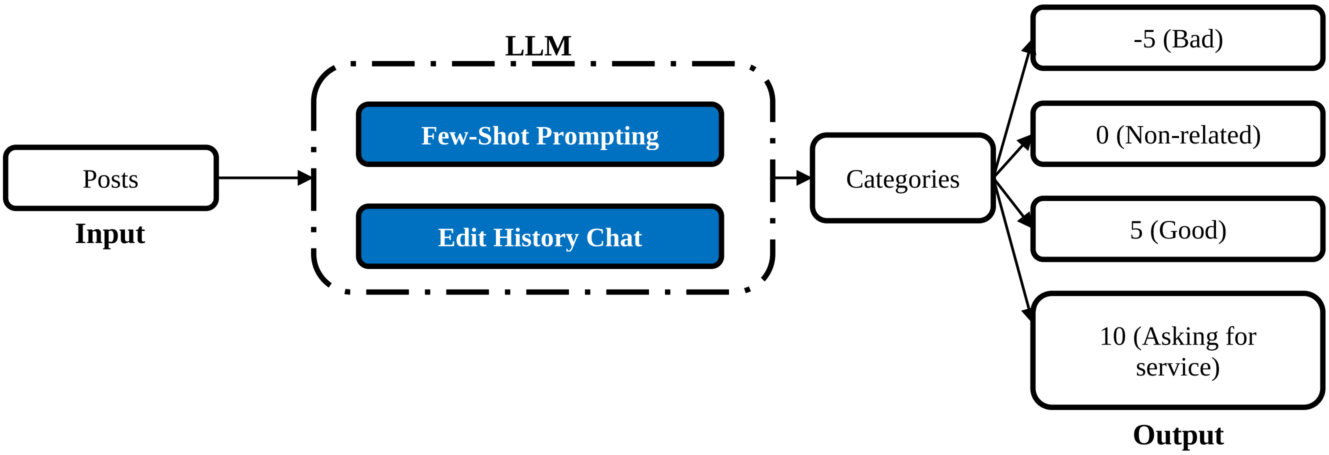

Fig. 2. Prompt techniques integrate into Feedback Evaluation process

IV. METHODOLOGY

This study, through an extensive review of relevant research, presents a meticulously designed architecture aimed at addressing the challenge of assessing reputation trust score in online social networks, as illustrated in the accompanying **Fig. 3**. Our architecture to solve the problem "Calculating reputation trust score on social networks" includes 3 main modules: Data Collection, Data Preprocessing, Trust Computation model, details of these modules are presented below.

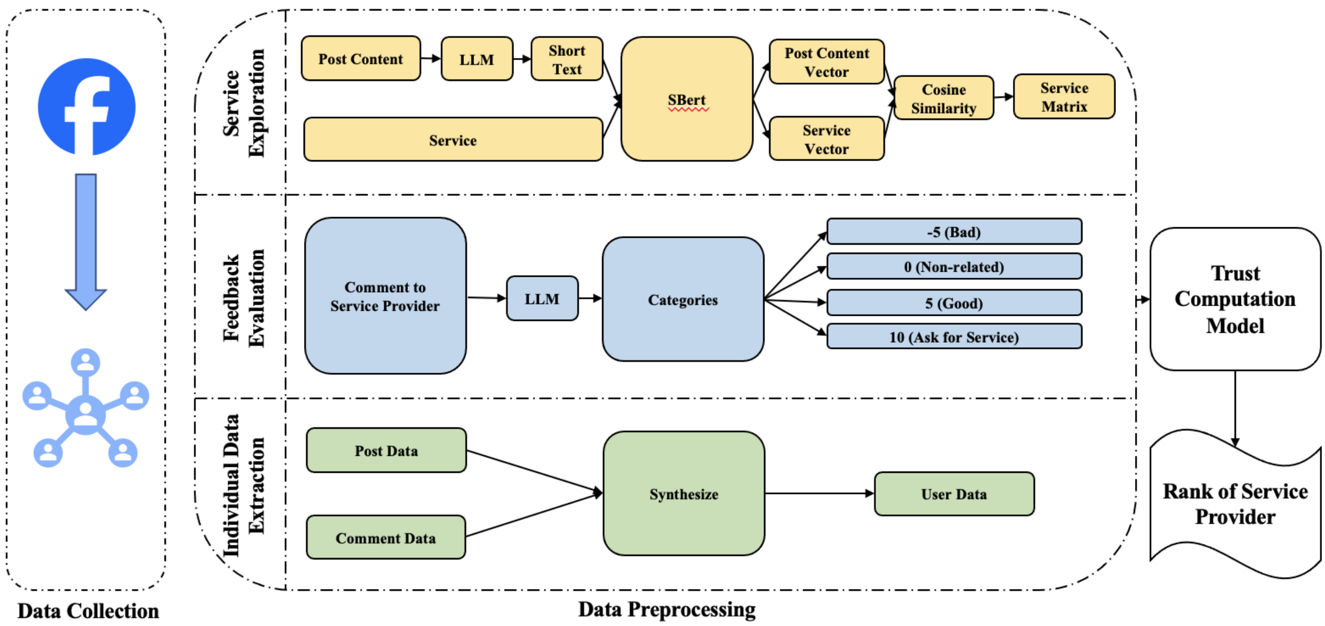

Fig. 3. Our architecture for calculating reputation trust score in social networks

A. Data Collection

The process and steps for collecting data in more detail will be mentioned in **Appendix D Section: Data Collection in Detail.** The dataset comprises information about posts and their associated comments. Post details include post ID, poster's ID, content, reactions received, and posting time. Comment details encompass comment ID, commenter's ID, content, posting time, and reactions received. Comments are linked to their respective posts through the post ID.

Let represent the original data of Facebook group as a graph:

$$G_f = \{U, P, C, R\} \tag{1}$$

Where U is the set of users in a social network as:

$$U = \{u_1, u_2, \dots, u_n\} \tag{2}$$

P contains the number of posts associated with each u_i in the group and represents as follow:

$$P = \{ P_1, P_2, \dots, P_n\} \tag{3}$$

P_i is the set of posts provided by user u_i, a user may have several posts:

$$P_i = \{p_{i,k} \mid k = 1,2,3, \dots, m\} \tag{4}$$

C contains the comment of u_i to post $p_{j,k}$ of u_j as follow:

$$C = \{ C_{i,j,k} \mid i = 1,2,3, \dots, n ; j = 1,2,3, \dots, n ; k = 1,2,3, \dots, m \} \tag{5}$$

R contains the reacts that user u_i received:

$$R = \{R_1, R_2, \dots, R_n\} \tag{6}$$

Where R_i is the number of reactions of each type that u_i receives:

$$R_i = \{r_{like}, r_{love}, r_{haha}, r_{care}, r_{wow}, r_{angry}, r_{sad}\} \tag{7}$$

B. Data Preprocessing

To prepare data suitable for the input of the trust computation model (the next module), the data collected in the Data Collection module must go through the following processing steps: Service Exploration, Feedback Evaluation, and Individual Data Extraction.

Service Exploration

This processing step helps identify service providers within the social network. To perform this step, information is extracted from posts using an LLM model to produce a short text summary about the service provided in the post, if any. For the content of posts P_i, which contains information related to a service a user is offering, we will extract information about the type of service it provides using the LLM model. Let S be the set of service providers on the social network with:

$$S = \{S_1, S_2, \dots, S_q\}, S \in U \tag{8}$$

To find service providers S from set of users U in the social network, the content in the posts P_i associated with user u_i will be passed into LLM to extract service related tag_i.

Where:

$$tag_i = \{t_1, t_2, \dots, t_q\} \tag{9}$$

t_i is the service associated with a post in P_i of u_i , to avoid redundancy, only separate post will be pass into the LLM by using Levenshtein distance [52] to filter duplicate post

The Levenshtein distance between two strings a, b, (of length |a| and length |b|) is given by $lev(a, b)$

$$lev\ (a, b) = \begin{cases} |a|\ if\ |b| = 0, \\ |b|\ if\ |a| = 0, \\ lev(tail(a), tail(b))\ if\ head(a) = head(b), \\ 1 + \min \begin{cases} lev(tail(a), b) \\ lev(a, tail(b)) \\ lev(tail(a), tail(b)) \end{cases} \quad otherwise \end{cases} \tag{10}$$

Where the $tail$ of some string x is a string of all but the first character of x and $head\ (x)$ is the first character of x.

The type of services needs to be survey is represented as

$$Ts = \{Ts_1, Ts_2, Ts_3, \dots, Ts_o\} \tag{11}$$

With each type of service Ts_i , to match with service mentioned in tag_i , a set of word $setTs_i$ will be created as follow:

$$setTs_i = \{word_1, word_2, \dots., word_v\} \tag{12}$$

And the vector embedded by SBert [53] model of any word is represented as:

$$embed(word) = \{a_i \in [0,1] | i = 1, \dots, 768\} \tag{13}$$

Finally, we will use cosine similarity to match number of service mentioned in tag_i with set of word $setTs_i$ after embedding step by SBert model, if $cosine(embed(t_i), embed(word_i)) > delta$ then $u_i \in S$. Where cosine similarity of vector a and b is represented as:

$$cosine(a, b) = \frac{a.b}{||a||.||b||} \tag{14}$$

Feedback Evaluation

This processing step helps to identify and assess the quality of a service provider's services. Comments on the service provider's posts will be analyzed using the LLM model to classify the comments into positive, negative, non-related and service request categories.

The weight for each type of comments is presented as follow:

$$TypeC = \{w_p, w_n, w_s, w_{non}\} \tag{15}$$

To evaluate the relationship between a user u_i to a service provider $u_j \in S$, all the comment associated with their weight will be sum up according to this formula:

$$E\ (i, j) = \sum_{h=0}^{k} C_{i,j,h} \times TypeC\ (C_{i,j,h}) \tag{16}$$

Individual Data Extraction

This processing step will help gather information related to the interactions of an individual within the social network. Information such as the number of posts, comments made, and interactions, as well as comments received by service providers, will also be compiled. These interaction indices are defined in **Data Collection.**

C. Trust Computation Model

After the data has been processed through Module C (Data Pre-processing), we will input it into a model to calculate trust. In this study, we propose an integrated trust calculation model that uses knowledge graphs and our model (an algorithm developed based on the Improved PageRank algorithm) to calculate trust for service providers. This model will help us give a score for the reliability of service providers based on their interaction with users then we will proceed to classify and rank the service providers based on the score calculated from the model. This model can be summarized in the following easy-to-understand steps:

- Model data in the form of knowledge graphs: data received from module 2 (Data Preprocessing) will be standardized into a matrix and then included in the knowledge graph.
- Calculate Trust score through TFT Model: in this module we use data after being processed in step 1, the data will be passed through our TFT model (an algorithm developed based on the Improved PageRank algorithm) to give the final trust score

Here we will present in more detail the main steps of calculating trust score, using data embedded in Knowledge Graphs and our TFT model (an algorithm developed based on the Improved PageRank algorithm).

Step 1: Model data in the form of knowledge graphs

After preprocessing the data, a knowledge graph will be constructed as follows:

$$G = \{V, E\} \tag{17}$$

Where V is the set of normal users and service providers in the social network that has the same distinct number of nodes as U.

$$V = N \cup S \tag{18}$$

N is the set containing normal users in the social network:

$$N = \{n_1, n_2, \dots, n_n\} \tag{19}$$

Where u_i has the following accompanying indices:

$$n_i = [R, C_{out}, C_{in}, P\ ] \tag{20}$$

C_{out} *and* C_{in} includes the number of comments given and received by n_i , the loop is not count as user cannot self-comment to promote their score.

Where:

$$C_{out} = \sum_{j=0}^{n} \sum_{h=0}^{k} C_{i,j,h.}\ ;\ C_{in} = \sum_{j=0}^{n} \sum_{h=0}^{k} C_{j,i,h} \tag{21}$$

S is the set containing service providers in the social network:

$$S = \{s_1, s_2, \dots, s_m\} \tag{22}$$

Where s_i also has the same indices as u_i .

E is the edge connecting user u_i and service provider s_j with weight w represented as follows:

$$E = \{e_{i,j,w} \mid i = 1,2,3 \ldots n, j = 1,2,3, \ldots, m, w \in R\} \tag{23}$$

Then, from the initial directed graph, it will be converted into an adjacency matrix A with a size equal to the total number of l distinct nodes in set V. Where:

$$A_{l \times l} = \{a_{i,j} \in R \mid i = 1,2,3, \ldots, l, j = 1,2,3, \ldots, l\} \tag{24}$$

Step 2: TFTmodel

In the traditional PageRank algorithm, every page starts with an identical rank, and all links are treated the same when distributing rank scores, resulting in many iterations to reach the final PageRank. To enhance this process, the proposed algorithm assigns initial Improved PageRank [54] values considering the number of in links and outlines each page has, prioritizing in links. This approach is based on the idea that "a page is significant if other significant pages link to it." This modification decreases the number of iterations needed, thus speeding up the calculation. In social networks, building relationships with influential users can elevate your status. The improved PageRank algorithm captures this essence; when a node connects to many important nodes, its value and influence increase. This concept applies to web pages and social networks, biological systems, and recommendation systems. Each node represents an individual or entity, and links represent relationships or interactions. More connections with influential people enhance visibility and message dissemination, strengthening one's position in the community. Identifying and connecting with key individuals in a social network can lead to sustainable growth, facilitating rapid information spread and benefiting personal branding, business development, and marketing.

Algorithm 1: TFTModel

Input: Knowledge Graph Embedding $G(V, E)$,
Output:
$S \leftarrow \emptyset$ //
// Caculate Interaction Trust
foreach $v \in V$ **do**
  $Itrust_v \leftarrow \frac{I_r}{I_p} \times \alpha + I_c \times \beta$
// Caculate Service Trust
foreach $v \in V$ **do**
  $Strust_{j(0)} \leftarrow \frac{2(2I_j + O_j)}{\sum_{K \in R(Pj} (I_k + O_k)}$
do
  foreach $v \in V$ **do**
    $Strust_{j(0)} \leftarrow (1 - d) + d(\sum_{j \in L_i} \frac{Strust_j}{O_j})$
  end for
while $Strust_{j(k)} = Strust_{j(k-1)}$
// Aggerate interaction trust and service trust
for each $v \in V$ do
  $FTScore_v \leftarrow Strust_v \times (1 + \frac{Itrust_v}{\sum_{j=1}^{0} Itrust_j})$
  $S.append(\{v: FTScore_v\})$
return S;

In this research, to capture the characteristics of social network, we propose $TFTscore$ for trust evaluation. The algorithm is described in **Algorithm 1**.

Specifically, $TFTScore$ is the service provider trust score calculated by the formula:

$$TFTscore_i = Strust_i \times (1 + \frac{Itrust_i}{\sum_{j=1}^{0} Itrust_j}) \quad (25)$$

Based on the improved PageRank algorithm by Bama et el [54], the initial service trust value will be initialized as follows:

$$Strust_{j(0)} = \frac{2(2I_j + O_j)}{\sum_{K \in R(Pj}(I_k + O_k)} \quad (26)$$

$Strust_{j(0)}$ is the service trust of service provider S_j ; Ij and Oj are the in-links and out-links of S_j in matrix A; I_k and O_k are the in-links and out-links of user U_k, $R(S_j)$ are the user that reference to service provider S_j.

For each page in $j = 0,1, \dots, n$

$$Strust_{j(0)} = \frac{2(2I_j + O_j)}{\sum_{K \in R(Pj}(I_k + O_k)} \quad (27)$$

For each page k, $k = 0,1, \dots, n$ calculate the service trust recursively:

$$Strust_{j(0)} = (1 - d) + d(\sum_{j \in L_i} \frac{Strust_j}{O_j}) \quad (28)$$

End for:

$$Strust_{j(k)} = Strust_{j(k-1)} \quad (29)$$

However, in addition to factors such as user interaction that affect the trust score of a service provider on social networks, there are still many other factors that can influence the scoring process. One of the biggest influencing factors is other users' feedback in service providers' posts about whether they order from that service provider or not. Additionally, the weight of each node is also aggregated from the indices R, C, P as follows:

$$Itrust_i = \frac{I_r}{I_p} \times \alpha + I_c \times \beta \quad (30)$$

Where:

$$I_r = r_{like} + r_{love} + r_{haha} + r_{care} + r_{wow} - r_{sad} - r_{angry} \quad (31)$$

$$I_c = C_{in} \quad (32)$$

$$I_p = P \quad (33)$$

D. Evaluation metrics

For our problem, here we will apply two evaluation methods: Kendall's Tau [55] and Monotonicity Relation [55] to calculate the correlation of the two algorithms. In addition, we will also compare the rank order results returned by the model with the rank order of service providers collected by us through a survey form. More details about the two algorithms Kendall's Tau and Monotonicity will be updated in the **Appendix E Section: Evaluation Metrics and Methods.**

V. RESULTS

We have collected a data set of interactions of users on the social network Facebook in a group named FU - Hoa Lac. The data set includes about 4,875 posts from 6,330 users and 58,587 comments. We collected the data using the Selenium library, which helps simulate user actions on Facebook, such as logging into an account, accessing the group, and browsing through posts. Selenium retrieves information by accessing the HTML of the post pages. However, due to Facebook's policies and privacy measures, requests from the same IP are monitored, and any large number of requests in a short period can result in the IP being blocked. This limitation restricted our data collection efforts, preventing us from gathering more data. After collection, our data underwent a preprocessing step before being converted into graph data, the detailed results of converting into graphs will be shown in **Table I**.

TABLE I. GRAPH INFORMATION IN DETAILS

Components	**Details**
Node Labels	1371 nodes (service providers/ users)
Relationship Types	1423 interactions
Property Keys	comment_receive, name, num_angries, num cares, num_hahas, num_likes, num_loves, num_sads, num_wows, owner, owner_name, score, self_comments, service_types, total_comment_reacts, total_comments, total_posts, weight

From the total 6,330 users, we identified 296 service providers. These services are divided into six main categories: an_uong (food), giat_la (laundry), van_chuyen (transport), giao_duc (education), my_pham (cosmetics), and various other services. As illustrated in the chart, food-related services dominate in FU - Hoa Lac, with about 78 providers, significantly outnumbering providers of other services.

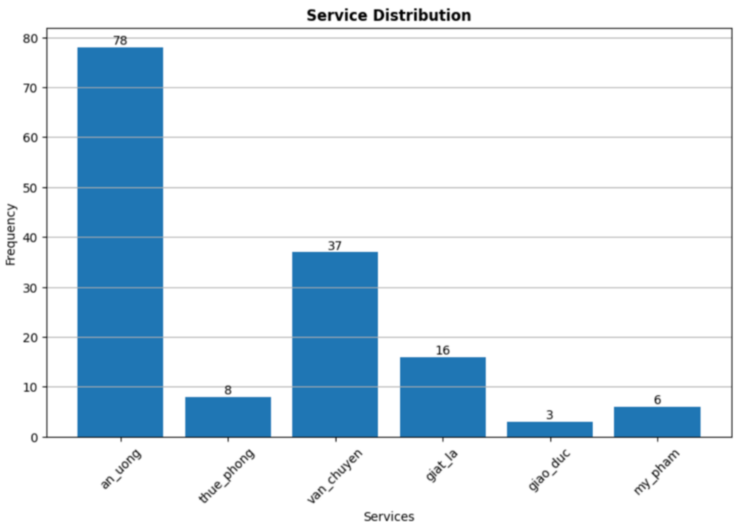

Fig. 4. *Frequency Distribution of Service Categories*

Although there are only about 296 service providers out of a total of 6330 users (accounting for about 4.7%), the amount of interaction they bring to the group is extremely outstanding. With only 4.7% of users, each type of interaction brings an average of 24.6% of the group's interactions. Looking at the chart, we can easily see that service providers tend to post a lot and comment on their posts a lot.

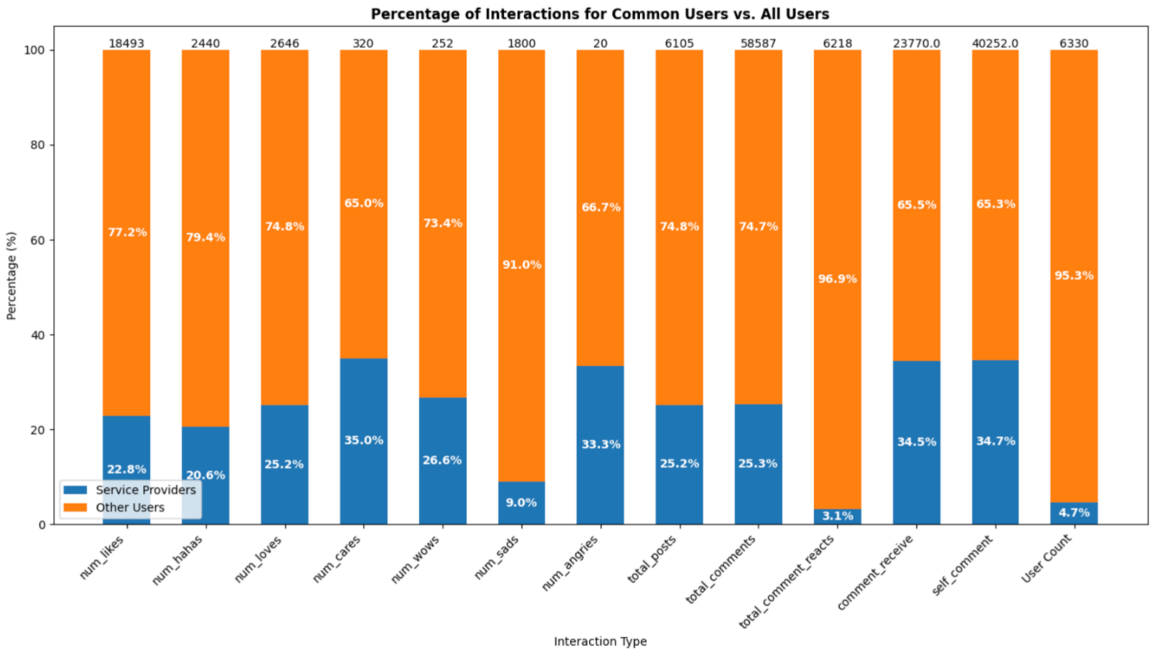

Fig. 5. Comparative Analysis of Interaction Types: Service Providers vs. Other Users

In addition, in this section we also explain to you how we choose values for the two parameters alpha and beta, more details will be updated in the **Appendix F Section: Additional Experimental Results.**

Next, the scores from the improve PageRank algorithm of each service provider are aggregated according to the formula () for the following representation:

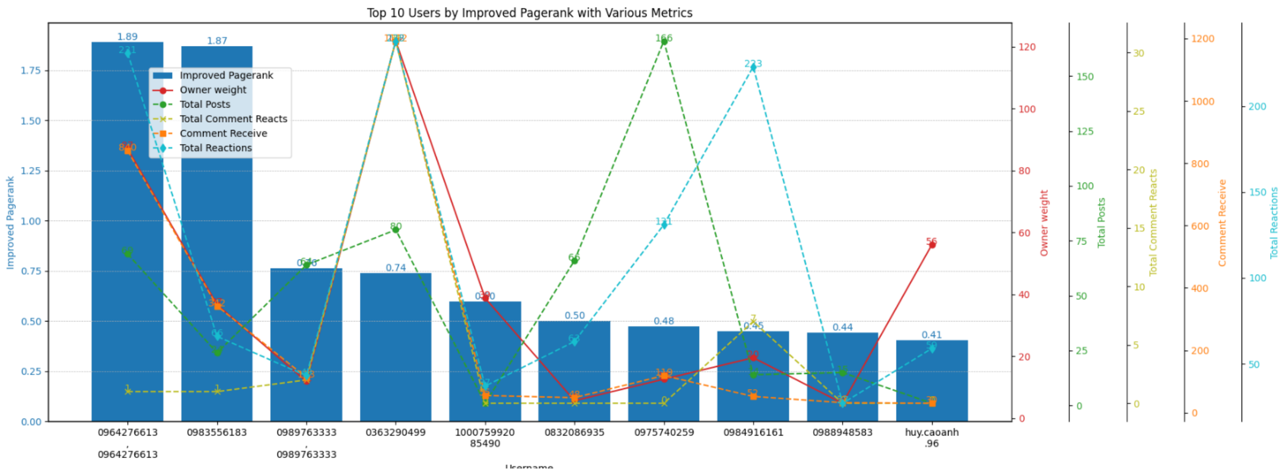

Fig. 6. Comparative Analysis of Top 10 Users by Improved PageRank and Interaction Metrics

As can be seen, the improved PageRank score has a non-linear relationship with interaction metrics such as reactions, comments, and the number of posts. This can be explained by the fact that the improved PageRank score is derived from the adjacency matrix. The adjacency matrix A is formed by the sum of weights between pairs of users, where the weights are aggregated from the content of comments according to formula (15). Therefore, the quality of interactions plays an important role in the PageRank score. A service provider may have many interactions from comments, posts, and reactions, but if there are not many comments placing orders or requesting services, the score between users and that service provider will not be high. The quality of service should also base on the engagement of that service provider in the social network, so we need to combine the score from Improved PageRank and user weight for the final trust result.

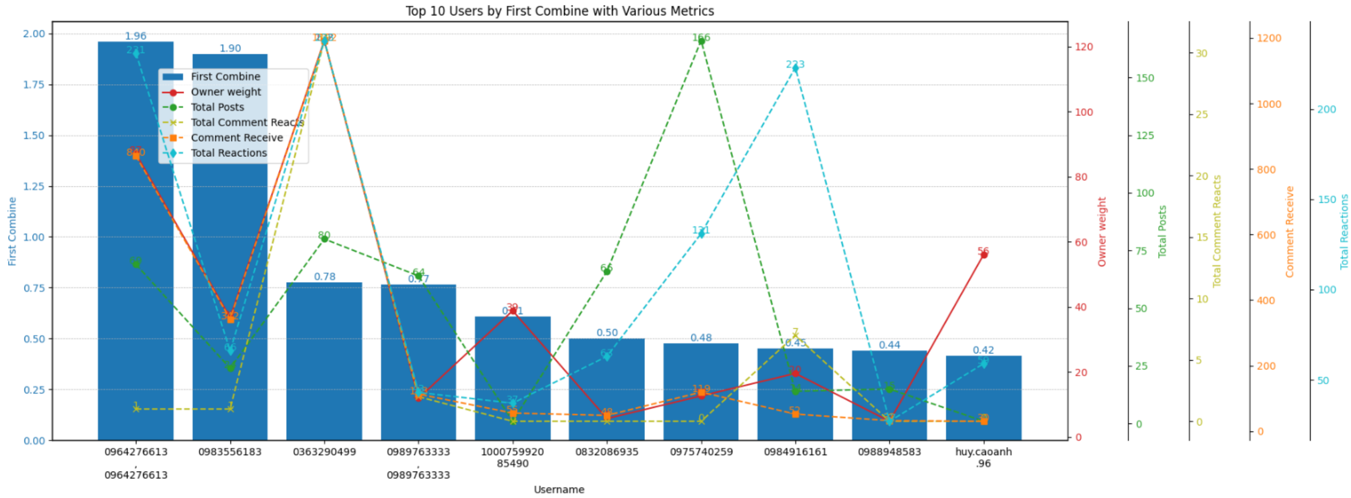

Fig. 7. Comparative Analysis of Top 10 Users by TFT Model and Interaction Metrics

Fig. 7 describes the interaction indicators of the top 10 most reputable service providers proposed by the TFTmodel algorithm. The number of interactions of service providers is often high, with many users posting up to nearly 200 posts along with The number of reactions is also quite large, on average, about 100 reactions. In addition, the number of comments received from these users is also quite large and are often comments ordering the service of the provider. The comments always receive positive feedback and a large number of orders. The most special point is that these are all food-related service providers.

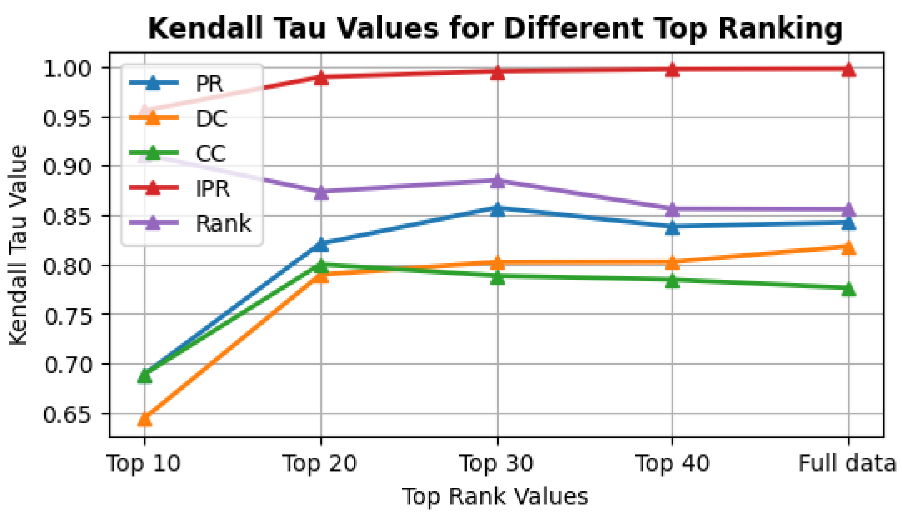

***Fig. 8.** Kendall's Tau algorithm for TFTModel value compared to other algorithms (purple-line: Survey Result, blue-line: PageRank, red-line: Improved PageRank, green-line: Closeness Centrality, orange-line: Degree centrality)*

To evaluate the results of TFTModel, we use the index Kendall's Tau to compare the results of the model with algorithms such as improved pagerank, page rank, degree centrality [55], closeness centrality [55] (These are the basic measures in calculating the scores of nodes in the graph). In addition, we also built a survey form about the quality of service providers we collected on the FU - Hoa Lac group, the survey was built for each service provider with a scale of 0. to 5 (from very bad to very good), we will use the results of the rankings of service providers from the survey to compare the rankings of service providers according to the TFTModel algorithm returned by Kendall's Tau index to give the most objective assessment, the results will be displayed in the ***Fig. 8***.

The ***Fig. 8*** evaluates the Kendall's Tau index between the TFTModel results we return compared to other types of algorithms as well as Results from the survey form according to top 10, top 20 and all data respectively. The model's Kendall's Tau evaluation index when compared with other classic models shows quite close results, especially with the improved page rank algorithm giving almost similar results (because my algorithm is based on the idea of improved page rank algorithm in determining the reputation trust score of service providers). Looking at the Kendall's Tau index between the survey form and the model, we can see that the value gradually decreases as the top value gets larger, so we can see that TFTModel has quite high accuracy for service providers but the accuracy decreases later. For the survey form, the top 10 returned results we reached 91.1%, for the top 20 models our accuracy compared to the survey form was about 87.3%, for the top 30 models it was about 88.5% and For all data, the result kendall's tau returns is 85.6%. To explain this, we can understand that the survey form here may not have fully collected the opinions of customers in the group, so the model may show discrepancies. However, the model returns quite accurate results for the top service providers, showing that our model is capable of recommending trustworthy service providers in the group. To continue evaluating the monotonicity of the models, I use the monotonicity index to evaluate the above algorithms. According to ***Fig. 9***, we can see the results of the TFTModel algorithm with service providers having the highest differentiation (from 0.95 to 0.99 corresponding to the top 10 to the entire data) and degree centrality being the lowest (from 0.75). to 0.89 corresponding to the top 10 to the entire data). However, compared to the results of the first top 20 nodes, TFTModel, improve page rank and page rank algorithms have equal indexes (this can be explained because these 3 algorithms were all developed from an original algorithm: page rank) but with the top 20 onwards there are differences, here the TFTModel algorithm produces better results than the remaining algorithms, showing that the model we developed is giving good results. Better results than the remaining classic models.

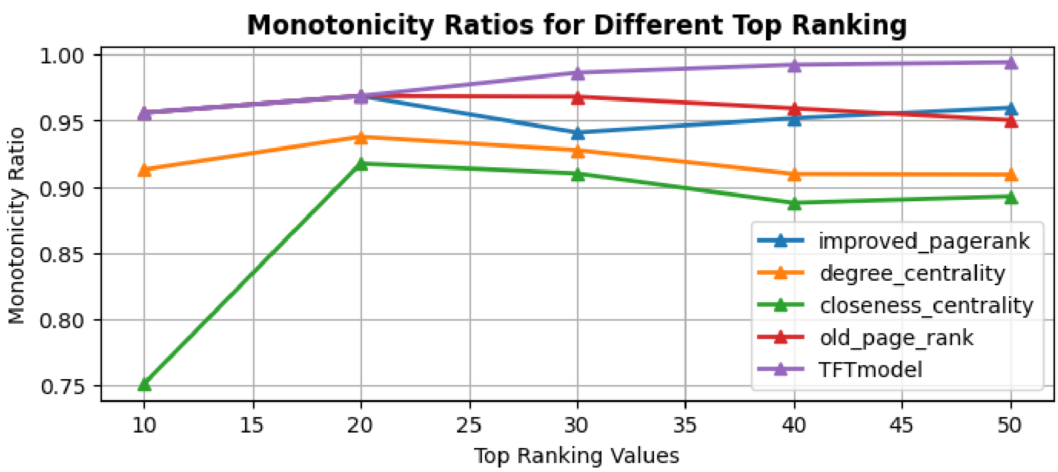

Fig. 9. *Monotonicity algorithm for TFTModel value compared to other algorithms (purple-line: TFTModel, red-line: PageRank, blue-line: Improved PageRank, orange-line: degree centrality: green-line: Closeness Centrality)*

Finally, to show that the model and problem we developed have high applicability in practice, we have built a website to evaluate the reliability of service providers in the HL group. Websites will display their trust score along with other metrics like number of interactions, profile information, and data returned from our algorithm. By using information from the web, we hope that users can better understand the services around them.

VI. CONCLUSION

In this study, we implemented applications of large language models (LLM) for contextual analysis in social networks, a field that requires a deep understanding of natural language and its complexity. complexity of social interactions. By applying LLM, we can analyze interactive content accurately and in detail, thereby providing valuable assessments and insights. We also apply Knowledge Graph (KG) in calculating Reputation Score and Trust Score. These two scores are combined with Interaction Score and Service Score based on improved PageRank algorithm. The set of weights between interactions is also optimized to ensure the accuracy and efficiency of the evaluation system. During data preprocessing, we built functions that remove loops and seed comments. This is done by eliminating users with the same ID or similar comments, ensuring that the data used is clean and of high quality. This helps to minimize the impact of confounding factors and improve the accuracy of the analysis. We compared the proposed model with classical models, and real-life survey results show that our model achieves remarkable accuracy. This proves that combining LLM and KG in analyzing social network contexts is a right and potential direction. In addition, we have built a web system and deployed this application for users in the FU Hoa Lac group to refer to and use. This web system is not only a useful tool but also a demonstration of the feasibility and effectiveness of the model we propose. Users can access and experience the system's features, providing valuable feedback for us to continue improving.

In the future, we will focus on several key research and development directions. First, we will continue to collect and expand the data set to increase diversity and coverage, making our model applicable to many different types of contexts and situations in social networks. Real-time data processing will also be promoted to improve system performance and interactivity, thereby providing immediate analysis and evaluation, meeting user needs quickly. fast and accurate. Besides, we will continue to research and improve the LLM model to solve complex context analysis problems in social networks. These improvements include the optimization of deep learning algorithms and the application of advanced machine learning techniques. We will also research and apply graph neural network (GNN) models to enhance the ability to analyze and process complex data in social networks. GNN will help us exploit relationships between entities in graphs more effectively, thereby improving the quality and accuracy of contextual analytics. This research not only opens new directions in social network context analysis but also provides a solid foundation for future research and applications. We hope that the results achieved will contribute to the development of this field and bring practical value to the social network user community.

REFERENCES

[1] J.-H. Cho, K. Chan, and S. Adali, 'A Survey on Trust Modeling', *ACM Comput. Surv.*, vol. 48, no. 2, pp. 1–40, Nov. 2015, doi: 10.1145/2815595.

[2] S. Nepal, W. Sherchan, and C. Paris, 'STrust: A Trust Model for Social Networks', in *2011IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications*, Nov. 2011, pp. 841–846. doi: 10.1109/TrustCom.2011.112.

[3] S. Ding, Z. Yue, S. Yang, F. Niu, and Y. Zhang, 'A Novel Trust Model Based Overlapping Community Detection Algorithm for Social Networks', *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 11, pp. 2101–2114, Nov. 2020, doi: 10.1109/TKDE.2019.2914201.

[4] F. E. Walter, S. Battiston, and F. Schweitzer, 'Personalised and Dynamic Trust in Social Networks'. arXiv, May 09, 2009. doi: 10.48550/arXiv.0902.1475.

[5] Y. Wang, G. Yin, Z. Cai, Y. Dong, and H. Dong, 'A trust-based probabilistic recommendation model for social networks', *J. Netw. Comput. Appl.*, vol. 55, pp. 59–67, Sep. 2015, doi: 10.1016/j.jnca.2015.04.007.

[6] D. E. Saputra, 'Defining Trust in Computation', in *2020 International Conference on Information Technology Systems and Innovation (ICITSI)*, Oct. 2020, pp. 161–166. doi: 10.1109/ICITSI50517.2020.9264918.

[7] W. Sherchan, S. Nepal, and C. Paris, 'A survey of trust in social networks', *ACM Comput. Surv.*, vol. 45, no. 4, pp. 1–33, Aug. 2013, doi: 10.1145/2501654.2501661.

[8] W.-L. Chang, A. N. Diaz, and P. C. K. Hung, 'Estimating trust value: A social network perspective', *Inf. Syst. Front.*, vol. 17, no. 6, pp. 1381–1400, Dec. 2015, doi: 10.1007/s10796-014-9519-0.

[9] L. Mui, M. Mohtashemi, and A. Halberstadt, 'A computational model of trust and reputation', in *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, Big Island, HI, USA: IEEE Comput. Soc, 2002, pp. 2431–2439. doi: 10.1109/HICSS.2002.994181.

[10] 'Survey on Computational Trust and Reputation Models | ACM Computing Surveys'. Accessed: Apr. 10, 2024. [Online]. Available: https://dl.acm.org/doi/abs/10.1145/3236008

[11] A. Herzig, E. Lorini, J. F. Hübner, and L. Vercouter, 'A logic of trust and reputation', *Log. J. IGPL*, vol. 18, no. 1, pp. 214–244, Feb. 2010, doi: 10.1093/jigpal/jzp077.

[12] J. Sabater-Mir and C. Sierra, 'Review on Computational Trust and Reputation Models', *Artif Intell Rev*, vol. 24, pp. 33–60, Sep. 2005, doi: 10.1007/s10462-004-0041-5.

[13] M. G. Ozsoy and F. Polat, 'Trust based recommendation systems', in *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, in ASONAM '13. New York, NY, USA: Association for Computing Machinery, Aug. 2013, pp. 1267–1274. doi: 10.1145/2492517.2500276.

[14] F. E. Walter, S. Battiston, and F. Schweitzer, 'A model of a trust-based recommendation system on a social network', *Auton. Agents Multi-Agent Syst.*, vol. 16, no. 1, pp. 57–74, Feb. 2008, doi: 10.1007/s10458-007-9021-x.

[15] E. Majd and V. Balakrishnan, 'A trust model for recommender agent systems', *Soft Comput.*, vol. 21, no.

2, pp. 417–433, Jan. 2017, doi: 10.1007/s00500-016-2036-y.

[16] K. Zhao and L. Pan, ‘A Machine Learning Based Trust Evaluation Framework for Online Social Networks’, in *2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications*, Beijing, China: IEEE, Sep. 2014, pp. 69–74. doi: 10.1109/TrustCom.2014.13.

[17] X. Chen, Y. Yuan, L. Lu, and J. Yang, ‘A Multidimensional Trust Evaluation Framework for Online Social Networks Based on Machine Learning’, *IEEE Access*, vol. 7, pp. 175499–175513, 2019, doi: 10.1109/ACCESS.2019.2957779.

[18] S. Liu, L. Zhang, and Z. Yan, ‘Predict Pairwise Trust Based on Machine Learning in Online Social Networks: A Survey’, *IEEE Access*, vol. 6, pp. 51297–51318, 2018, doi: 10.1109/ACCESS.2018.2869699.

[19] M. Naderan, E. Namjoo, and S. Mohammadi, ‘Trust Classification in Social Networks Using Combined Machine Learning Algorithms and Fuzzy Logic’, *Iran. J. Electr. Electron. Eng.*, vol. 15, Feb. 2019, doi: 10.22068/IJEEE.15.3.294.

[20] S. Vahabli and R. Ravanmehr, ‘A novel trust-based access control for social networks using fuzzy systems’, *World Wide Web*, vol. 22, no. 6, pp. 2241–2265, Nov. 2019, doi: 10.1007/s11280-019-00668-y.

[21] F. Matinfar, ‘A Computational Model for Measuring Trust in Mobile Social Networks Using Fuzzy Logic’, *Int. J. Autom. Comput.*, vol. 17, no. 6, pp. 812–821, Dec. 2020, doi: 10.1007/s11633-020-1232-5.

[22] K. Su, B. Xiao, B. Liu, H. Zhang, and Z. Zhang, ‘TAP: A personalized trust-aware QoS prediction approach for web service recommendation’, *Knowl.-Based Syst.*, vol. 115, pp. 55–65, Jan. 2017, doi: 10.1016/j.knosys.2016.09.033.

[23] M. Parvathy, S. K., and S. Shalinie, ‘Cosine similarity-based clustering and dynamic reputation trust aware key generation scheme for trusted communication on social networking’, *J. Stat. Comput. Simul.*, vol. 85, pp. 1–12, May 2014, doi: 10.1080/00949655.2014.964240.

[24] Y. Ruan and A. Durresi, ‘A survey of trust management systems for online social communities – Trust modeling, trust inference and attacks’, *Knowl.-Based Syst.*, vol. 106, pp. 150–163, Aug. 2016, doi: 10.1016/j.knosys.2016.05.042.

[25] V. Podobnik, D. Striga, A. Jandras, and I. Lovrek, ‘How to calculate trust between social network users?’, in *SoftCOM 2012, 20th International Conference on Software, Telecommunications and Computer Networks*, Sep. 2012, pp. 1–6. Accessed: Apr. 11, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6347635

[26] Z. Bo, Z. Huan, L. Meizi, Z. Qin, and H. Jifeng, ‘Trust Traversal: A trust link detection scheme in social network’, *Comput. Netw.*, vol. 120, pp. 105–125, Jun. 2017, doi: 10.1016/j.comnet.2017.04.016.

[27] J. Wu, J.-L. Chang, Q. Cao, and C. Liang, ‘A trust propagation and collaborative filtering based method for incomplete information in social network group decision making with type-2 linguistic trust’, *Comput. Ind. Eng.*, vol. 127, Nov. 2018, doi: 10.1016/j.cie.2018.11.020.

[28] J. Wu, F. Chiclana, and E. Herrera-Viedma, 'Trust based consensus model for social network in an incomplete linguistic information context', *Appl. Soft Comput.*, vol. 35, pp. 827–839, Oct. 2015, doi: 10.1016/j.asoc.2015.02.023.

[29] Y. Wang, X. Wang, J. Tang, W. Zuo, and G. Cai, 'Modeling Status Theory in Trust Prediction', *Proc. AAAI Conf. Artif. Intell.*, vol. 29, no. 1, Feb. 2015, doi: 10.1609/aaai.v29i1.9460.

[30] S. M. Ghafari, S. Yakhchi, A. Beheshti, and M. Orgun, 'Social Context-Aware Trust Prediction: Methods for Identifying Fake News', in *Web Information Systems Engineering – WISE 2018*, H. Hacid, W. Cellary, H. Wang, H.-Y. Paik, and R. Zhou, Eds., Cham: Springer International Publishing, 2018, pp. 161–177. doi: 10.1007/978-3-030-02922-7_11.

[31] 'DCAT | Proceedings of the 17th International Conference on Advances in Mobile Computing & Multimedia'. Accessed: Apr. 11, 2024. [Online]. Available: https://dl.acm.org/doi/abs/10.1145/3365921.3365940

[32] 'A Survey on Trust Prediction in Online Social Networks | IEEE Journals & Magazine | IEEE Xplore'. Accessed: Apr. 11, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/9142365

[33] B. Qureshi, G. Min, and D. Kouvatsos, 'Trusted information exchange in peer-to-peer mobile social networks', *Concurr. Comput. Pract. Exp.*, vol. 24, no. 17, pp. 2055–2068, 2012, doi: 10.1002/cpe.1837.

[34] W. X. Zhao *et al.*, 'A Survey of Large Language Models'. arXiv, Nov. 24, 2023. doi: 10.48550/arXiv.2303.18223.

[35] S. Minaee *et al.*, 'Large Language Models: A Survey'. arXiv, Feb. 20, 2024. doi: 10.48550/arXiv.2402.06196.

[36] A. Baytak, 'The Content Analysis of the Lesson Plans Created by ChatGPT and Google Gemini', *Res. Soc. Sci. Technol.*, vol. 9, no. 1, Art. no. 1, Mar. 2024, doi: 10.46303/ressat.2024.19.

[37] R. Chew, J. Bollenbacher, M. Wenger, J. Speer, and A. Kim, 'LLM-Assisted Content Analysis: Using Large Language Models to Support Deductive Coding'. arXiv, Jun. 23, 2023. doi: 10.48550/arXiv.2306.14924.

[38] L. Giray, 'Prompt Engineering with ChatGPT: A Guide for Academic Writers', *Ann. Biomed. Eng.*, vol. 51, no. 12, pp. 2629–2633, Dec. 2023, doi: 10.1007/s10439-023-03272-4.

[39] 'Prompt Engineering in Large Language Models | SpringerLink'. Accessed: Apr. 12, 2024. [Online]. Available: https://link.springer.com/chapter/10.1007/978-981-99-7962-2_30

[40] Z. Li, X. Zhang, H. Shen, W. Liang, and Z. He, 'A Semi-Supervised Framework for Social Spammer Detection', in *Advances in Knowledge Discovery and Data Mining*, T. Cao, E.-P. Lim, Z.-H. Zhou, T.-B. Ho, D. Cheung, and H. Motoda, Eds., Cham: Springer International Publishing, 2015, pp. 177–188. doi: 10.1007/978-3-319-18032-8_14.

[41] S. M. Ghafari, S. Yakhchi, A. Beheshti, and M. Orgun, 'Social Context-Aware Trust Prediction: Methods for Identifying Fake News', in *Web Information Systems Engineering – WISE 2018*, H. Hacid, W. Cellary, H. Wang, H.-Y. Paik, and R. Zhou, Eds., Cham: Springer International Publishing, 2018, pp. 161–177. doi: 10.1007/978-3-030-02922-7_11.

[42] D. R. Bild, Y. Liu, R. P. Dick, Z. M. Mao, and D. S. Wallach, 'Aggregate Characterization of User Behavior in Twitter and Analysis of the Retweet Graph', *ACM Trans. Internet Technol.*, vol. 15, no. 1, pp. 1–24, Mar. 2015, doi: 10.1145/2700060.

[43] N. A. Abdullah, D. Nishioka, Y. Tanaka, and Y. Murayama, 'Why I Retweet? Exploring User's Perspective on Decision-Making of Information Spreading during Disasters', Jan. 2017. doi: 10.24251/HICSS.2017.053.

[44] X. Ma, H. Lu, and Z. Gan, 'Implicit Trust and Distrust Prediction for Recommender Systems', in *Web Information Systems Engineering – WISE 2015*, J. Wang, W. Cellary, D. Wang, H. Wang, S.-C. Chen, T. Li, and Y. Zhang, Eds., Cham: Springer International Publishing, 2015, pp. 185–199. doi: 10.1007/978-3-319-26190-4_13.

[45] Y. Yu, Y. Gao, H. Wang, and R. Wang, 'Joint user knowledge and matrix factorization for recommender systems', *World Wide Web*, vol. 21, no. 4, pp. 1141–1163, Jul. 2018, doi: 10.1007/s11280-017-0476-7.

[46] A. Caliò and A. Tagarelli, 'Complex influence propagation based on trust-aware dynamic linear threshold models', *Appl. Netw. Sci.*, vol. 4, no. 1, Art. no. 1, Dec. 2019, doi: 10.1007/s41109-019-0124-5.

[47] B. Abu-Salih *et al.*, 'Time-aware domain-based social influence prediction', *J. Big Data*, vol. 7, no. 1, p. 10, Feb. 2020, doi: 10.1186/s40537-020-0283-3.

[48] S. Sagar, A. Mahmood, K. Wang, Q. Z. Sheng, and W. E. Zhang, 'Trust-SIoT: Towards Trustworthy Object Classification in the Social Internet of Things'. arXiv, May 03, 2022. doi: 10.48550/arXiv.2205.03226.

[49] Y. Dai, S. Wang, N. N. Xiong, and W. Guo, 'A Survey on Knowledge Graph Embedding: Approaches, Applications and Benchmarks', *Electronics*, vol. 9, no. 5, Art. no. 5, May 2020, doi: 10.3390/electronics9050750.

[50] T. Ahmed, K. Pai, P. Devanbu, and E. Barr, *Improving Few-Shot Prompts with Relevant Static Analysis Products*. 2023.

[51] T. Wu *et al.*, 'PromptChainer: Chaining Large Language Model Prompts through Visual Programming', in *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems*, in CHI EA '22. New York, NY, USA: Association for Computing Machinery, Apr. 2022, pp. 1–10. doi: 10.1145/3491101.3519729.

[52] L. Yujian and L. Bo, 'A Normalized Levenshtein Distance Metric', *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 1091–1095, Jun. 2007, doi: 10.1109/TPAMI.2007.1078.

[53] N. Reimers and I. Gurevych, 'Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks', Aug. 27, 2019, *arXiv*: arXiv:1908.10084. doi: 10.48550/arXiv.1908.10084.

[54] '(PDF) Improved PageRank Algorithm for Web Structure Mining'. Accessed: Jul. 25, 2024. [Online]. Available:https://www.researchgate.net/publication/324987502_Improved_PageRank_Algorithm_for_Web_Structure_Mining

[55] P. Van Duong, X. T. Dinh, L. H. Son, and P. Van Hai, 'Enhancement of Gravity Centrality Measure Based on Local Clustering Method by Identifying Influential Nodes in Social Networks', Oct. 2022. doi: 10.1007/978-3-031-18123-8_48.

[56] T. Nguyen *et al.*, 'CulturaX: A Cleaned, Enormous, and Multilingual Dataset for Large Language Models in 167 Languages', Sep. 17, 2023, *arXiv*: arXiv:2309.09400. doi: 10.48550/arXiv.2309.09400.

APPENDIX A

SOME RELATED MODELS IN TRUST CALCULATION

In the context of online social networks, the participants are very diverse. Estimating the trustworthiness of each participant in the social network is necessary. Currently, many studies have been proposed to estimate the trustworthiness of these individuals. Details of some research directions are listed in **Table I**.

TABLE I. TRUST CALCULATION MODELS IN RELATED STUDIES

Research	**Approaches**	**Trust Formation**	**Trust Source**	**Trust Evaluation**
Nepal et al [2]	Reputation Similarity	Direct Indirect	Relationship Interactions Hybrid	Rank-based
Ding et al [3]	Reputation Similarity Comprehensive	Direct Indirect	Relationship Interactions Hybrid	Topology Cohesion
Walter et al [4]	Reputation Similarity Comprehensive	Direct Indirect	Relationship Interactions Hybrid	Rank-based
Wang et al [5]	Reputation Similarity Comprehensive	Direct	Relationship Interactions Hybrid	Rank-based
Chang et al [8]	Reputation Similarity Comprehensive	Direct Indirect	Relationship Interactions Hybrid	None
Zhao et al [16]	Reputation Similarity Comprehensive	Indirect	Relationship Interactions Hybrid	Confusion Matrix
Vahabli et al [20]	Reputation Similarity Fuzzy Comprehensive	Direct	Interactions Hybrid	AE MAPE

Wang et al [29]	Reputation Similarity Comprehensive	Direct Indirect	Relationship Interactions Hybrid	Rank-based

From **Table 1**, we see that the evaluation of each individual's trustworthiness typically relies on their relationships and interactions.

APPENDIX B

VISTRAL-7B-CHAT VIETNAMESE LLM MODEL

Vistral-7B-chat model [56] is a multi-way conversational large language model for Vietnamese. Vistral is an extension of the Mistral 7B model that uses diverse data to continuously pre-train and adjust instructions, Vistral's development process according to the author of this model includes 3 steps:

- Extend the tokenizer of Mistral 7B to better support Vietnamese.
- Perform continuous pre-training for Mistral over a diverse dataset of Vietnamese texts that are meticulously cleaned and deduplicated.
- Perform supervised fine-tuning for the model using diverse instruction data. We design a set of instructions to align the model with the safety criteria in Vietnam.

The model's author evaluated his Vistral model using the VMLU ranking, a reliable framework for evaluating large language models in Vietnamese on a variety of tasks. These tasks involve multiple-choice questions in STEM, Humanities, Social Sciences, and more. Our model achieves an average score of 50.07%, significantly surpassing ChatGPT's performance of 46.33%.

TABLE II. COMPARE THE PERFORMANCE OF VISTRAL-7B-CHAT WITH OTHER MODELS

VMLU Benchmark										
#	**Model**	**Creator**	**Access**	**Base Model**	**Evaluation Date**	**Stem**	**Social Science**	**Humanities**	**Others**	**AVG**
1	GPT-4	OpenAI	API	From Scratch	08/01/2024	63.4	71.78	66.14	60.37	65.53
2	Vistral-7B-Chat	UONLP x Ontocord	Weight	Mistral-7B-v0.1	16/01/2024	43.3	57.02	55.12	48.01	50.0
3	ChatGPT	OpenAI	API	From Scratch	08/01/2024	43.2	51.67	46.96	46.32	46.3
4	ViGPT-1.6B-v1	Vin BigData	Private	From Scratch	08/01/2024	35.0	48.72	47.20	42.54	42.3
5	Bloomz-7b1	BigScience	Weight	Bloom-7b1	08/01/2024	32.6	45.73	41.85	39.89	38.8

Appendix C

Prompt engineering techniques

These two methods are also the ones we apply in this problem. We will provide a clearer explanation of the characteristics and limitations of these two methods in **Table III**:

TABLE III. COMPARISON TABLE OF FEW-SHOT PROMPTING AND PROMPT CHAINING

Methods	**Descriptions**	**Limitations**
Few-shot-Prompting	This technique involves the user providing the LLM with a small amount of information (usually a few examples) before asking the model to perform a specific task.	Unable to handle overly complex tasks.
Prompt Chaining	This technique where a sequence of commands is designed to guide the model from an initial input state to a desired final result. Each command in the sequence is intended to build on the response of the previous command, allowing for the gathering of further information or directing the model to a more precise and detailed response.	Need a long command prompt

To help everyone better understand the two methods of prompting, Few-Shot Prompting and Prompt Chaining (with examples using ChatGPT 4), we will review examples of how these two methods are used in **Fig. 1.** and **Fig. 2.:**

- Few-Shot Prompting:

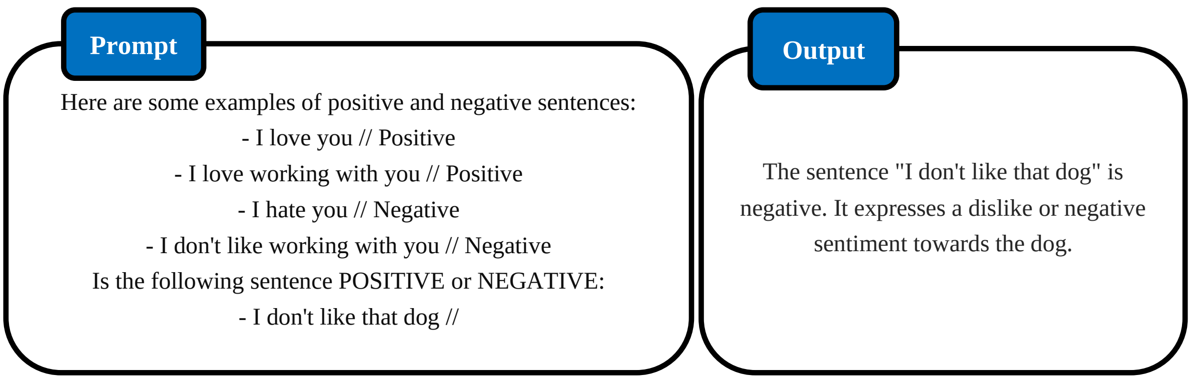

Fig. 1. *Example about using few-shot prompting*

- Prompt Chaining:

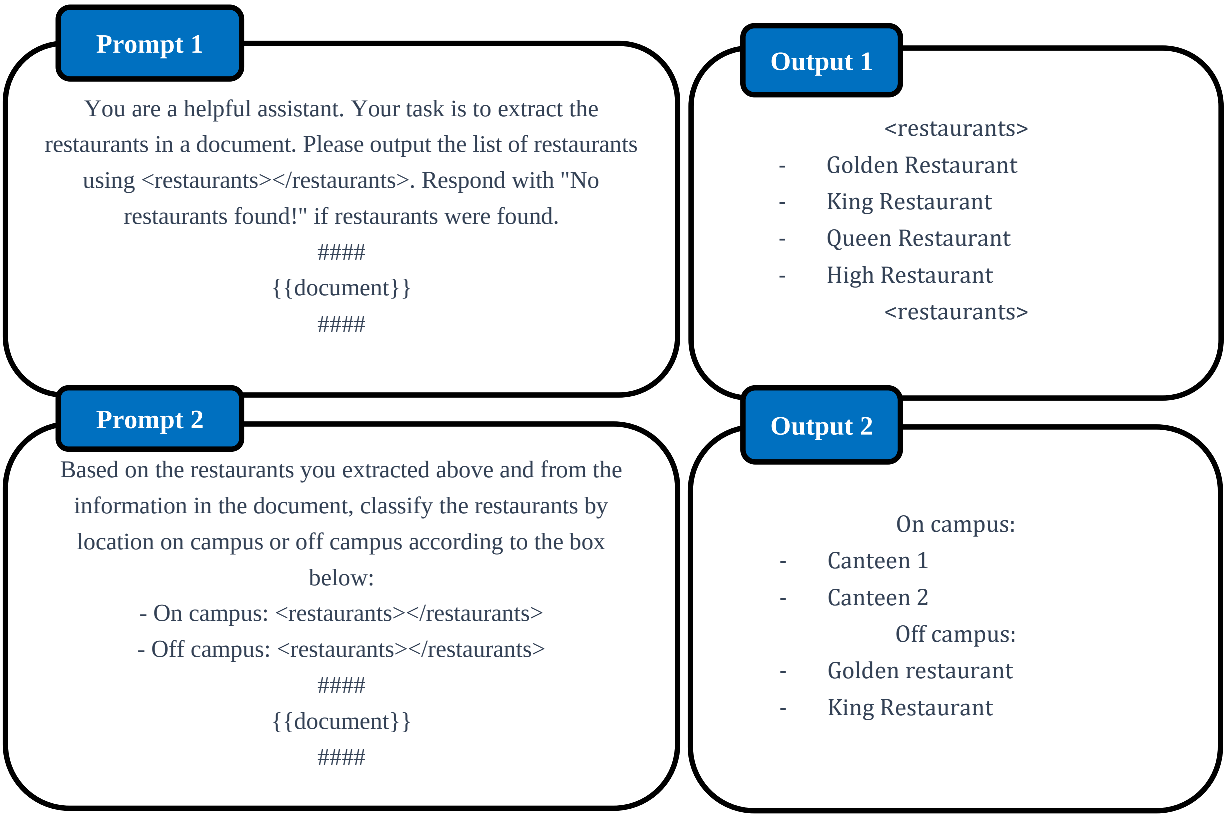

Fig. 2. *Example about using Prompt Chaining*

APPENDIX D

DATA COLLECTION IN DETAIL

The data used in the experiment was collected from a Facebook group named **FU-Hoa Lac**[1]. The library used is Selenium (version 4.19.0) – a Python library that helps users create bots to simulate real user actions on browsers, thereby making it easier to collect data from highly interactive websites like Facebook. The data collection process can be summarized in the following steps:

Step 1: Initialize the browser session (using Selenium Web-driver to initiate the Chrome browser)

Step 2: Access the Facebook page (Web-driver will then navigate to the required Facebook group)

Step 3: Search and collect data (using Selenium commands to retrieve necessary information such as content, comments, etc., this collection is conducted through the HTML of the website)

Step 4: Process and store data (after data collection, it will be cleaned and standardized into a CSV format for use).

[1] https://www.facebook.com/groups/356018761148436

Appendix E

Evaluation metrics and methods

Kendall's Tau

The Kendall Tau coefficient is an important index to determine the linear correlation between two series of numbers. The Kendall Tau coefficient has a value ranging from -1 to 1. The closer the absolute value of the Kendall coefficient is to 1, the higher the linear correlation between the two series of numbers. If the Kendall coefficient is 0, the two series of numbers do not have a linear correlation.

To calculate the Kendall Tau coefficient between two series of numbers $X = (x_1, x_2, x_3, \ldots, x_N)$ và $Y = (y_1, y_2, y_3, \ldots, y_N)$ with N elements:

$$T = \frac{2 \times (n_+ - n_-)}{N \times (N-1)} \qquad (36)$$

Where n_+ is the number of concordant pairs; n_- is the number of discordant pairs; N is the total number of pairs elements. Two pairs of elements (x_i, x_j) and (y_i, y_j) is concordant if $x_i > x_j$ and $y_i > y_j$ or $x_i < x_j$ and $y_i < y_j$. Conversely, two pairs of elements are discordant if $x_i > x_j$ and $y_i < y_j$ or $x_i < x_j$ and $y_i > y_j$.

Monotonicity Relation

The Monotonicity index $M(R)$ is used to measure the monotonicity of a ranked list. This index can be calculated according to the following formula:

$$M(R) = \left[1 - \frac{\sum_{r \in R} N_r \times (N_r - 1)}{N \times (N-1)}\right]^2 \qquad (37)$$

Where N is the size of the rating vector R; N_r is the number of nodes with the same rank value r. This metric measures the percentage of nodes with the same rank value in the rank list. $M(R)$ is in the range [0,1]. If $M(R) = 1$, the ranking algorithm is completely monotonic and each node is classified using a different index value. If $M(R) = 0$, all nodes have the same rank value. A higher M value indicates greater discrimination and uniformity of the ranked list.

Appendix F

Additional Experimental Result

In this section we will study the relationship between α and β values. We experimented with pairs of corresponding alpha and beta values: (0.9; 0.1); (0.8; 0.2); (0.7; 0.3); (0.6; 0.4); (0.5; 0.5); (0.4; 0.6); (0.3; 0.7); (0.2; 0.8); (0.1; 0.9). The standard deviation statistics between each user's score and each pair of values tested can be interpreted according to **Table V**. As in the case of using the value set (0.1; 0.9) when compared to the case of using the value set (0.9; 0.1), we can see that the case of using the value set (0.9; 0.1) gives us the value more stable.

TABLE IV. STATISTICS ON THE STANDARD DEVIATION BETWEEN USER'S SCORE ACCORDING TO EACH PAIR OF VALUE

	0.1-0.9	0.2-0.8	0.3-0.7	0.4-0.6	0.5-0.5	0.6-0.4	0.7-0.3	0.8-0.2	0.9-0.1
Standard Deviation	24.2233	21.7637	19.3899	17.1381	15.0629	13.248	11.8123	10.9084	**10.6717**

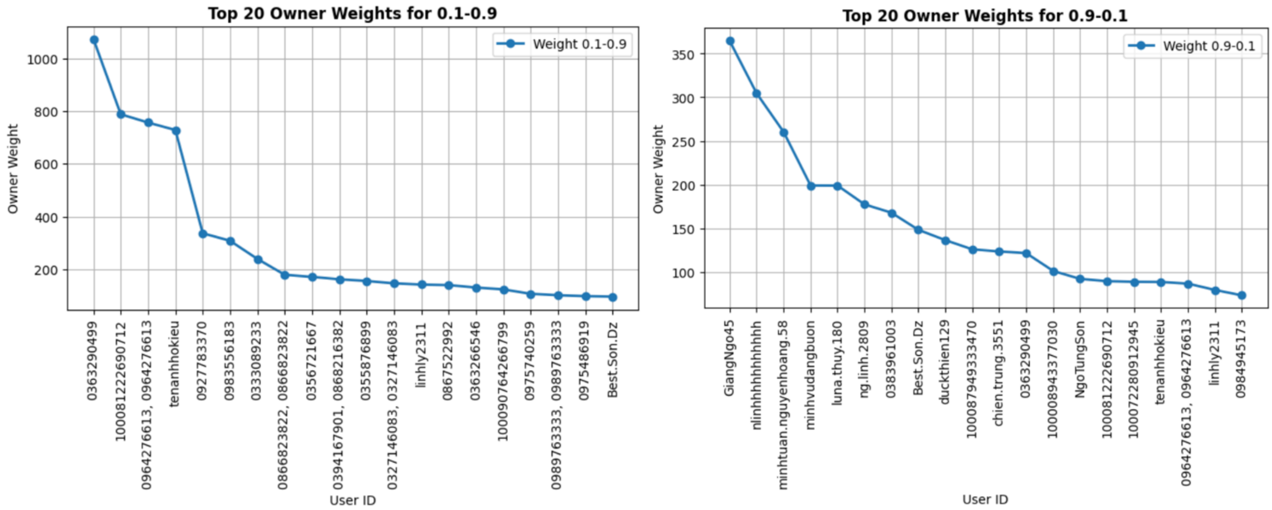

Fig. 3. *Top 10 Owner Weights Across Different α and β Ranges*

TABLE I. TABLE TYPE STYLES

Monday	**Tuesday**	**Wednesday**	**Thursday**	**Friday**	**Saturday**	**Sunday**