Fpt University
ĐẠI HỌC FPT

FPT Education - Research Festival 2024
Information Technology Session

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

TEAM: Trust Management

Agenda

1. Introduction

2. Methodology

3. Experiments & Result

4. Demo

1. Introduction

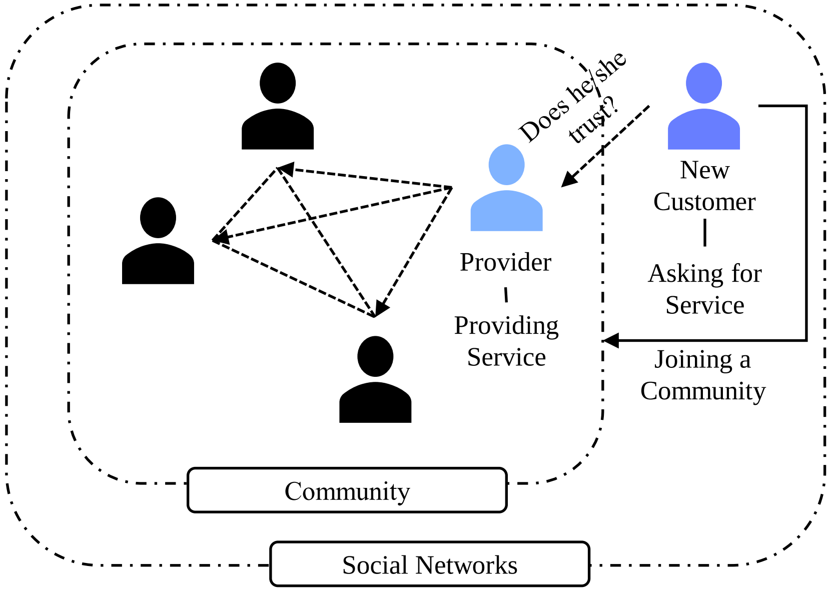


Figure 1. Why trust in Social Networks?

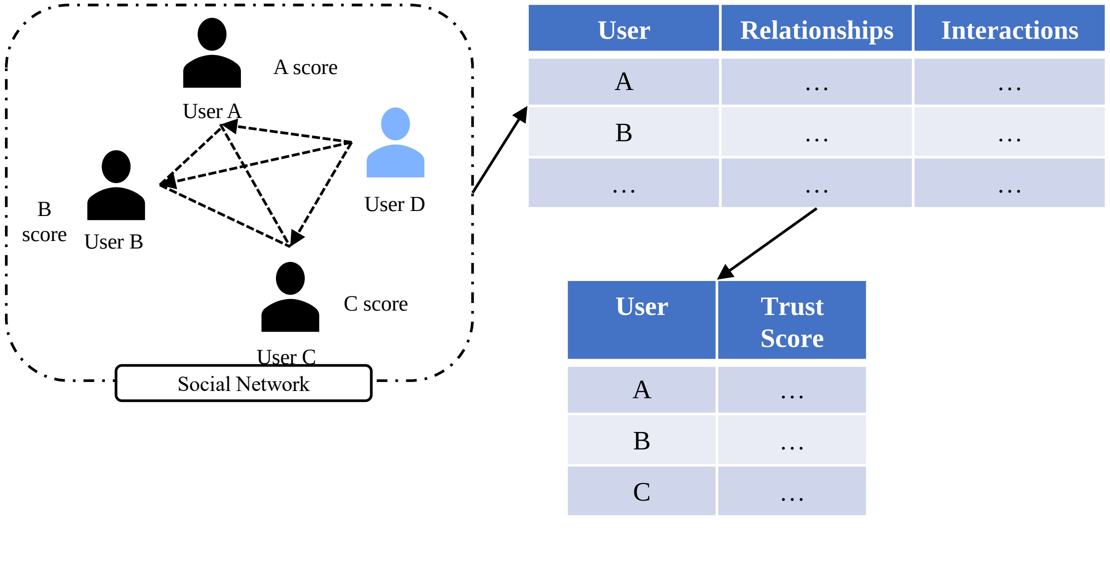


Figure 2. Defining Trust in Social Networks

- Social networks are no longer only a place for us to chat with each other every day, but it is also **a place where exchanges about goods take place.**
- Each person's trust in social networks becomes **extremely important**.
- Trust score is a score that represents a **person's trust in a social network based on that person's interactions and relationships on the social network**.

What are their problems?

- Old methods do not consider context in social networks.
- The nodes in the problem in the old method are equal and this is not practical.
- Using traditional deep learning models requires large amounts of data along with labels for that data.
- Evaluate user interactions in an unstructured and context-dependent data environment.

Our contributions

- Applying 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, self posts and comments.
- Building a TFT Model 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.
- Building a website to evaluate the trust scores of service providers on social networks.
- Building a core engine to crawl data from Facebook.

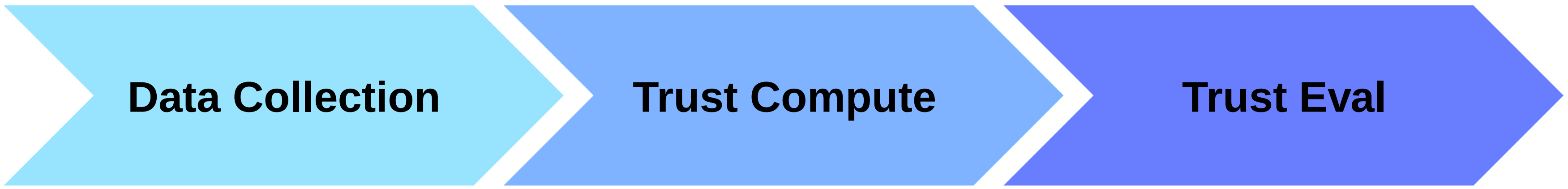
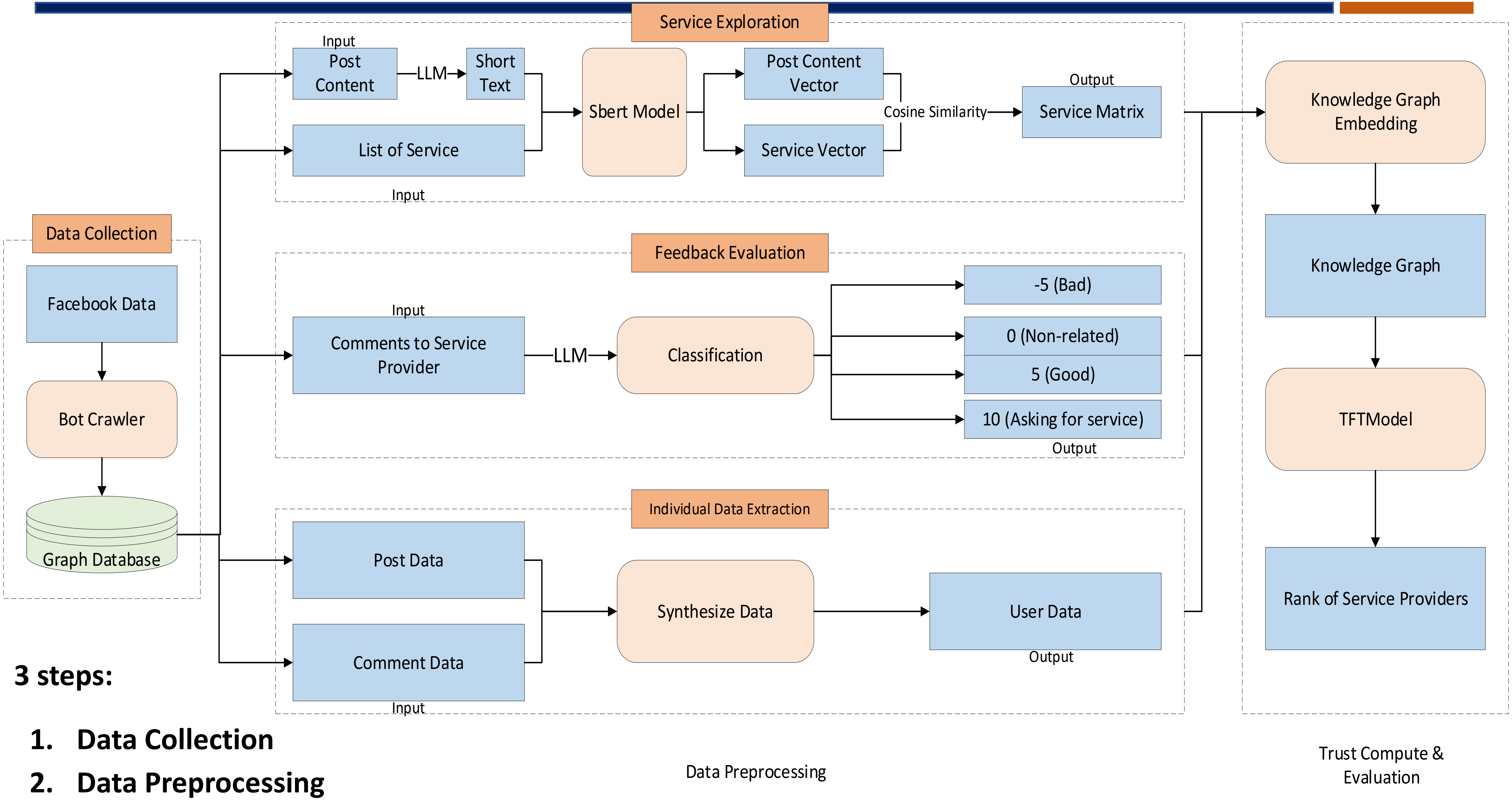


Figure 4. Basic Architecture of Trust Computation model

2. Methodology



3 steps:

1. **Data Collection**
2. **Data Preprocessing**
3. **Trust Compute & Evaluation**

Figure 5. Our architecture for Trust computation in social network

2. Methodology – Data Collection

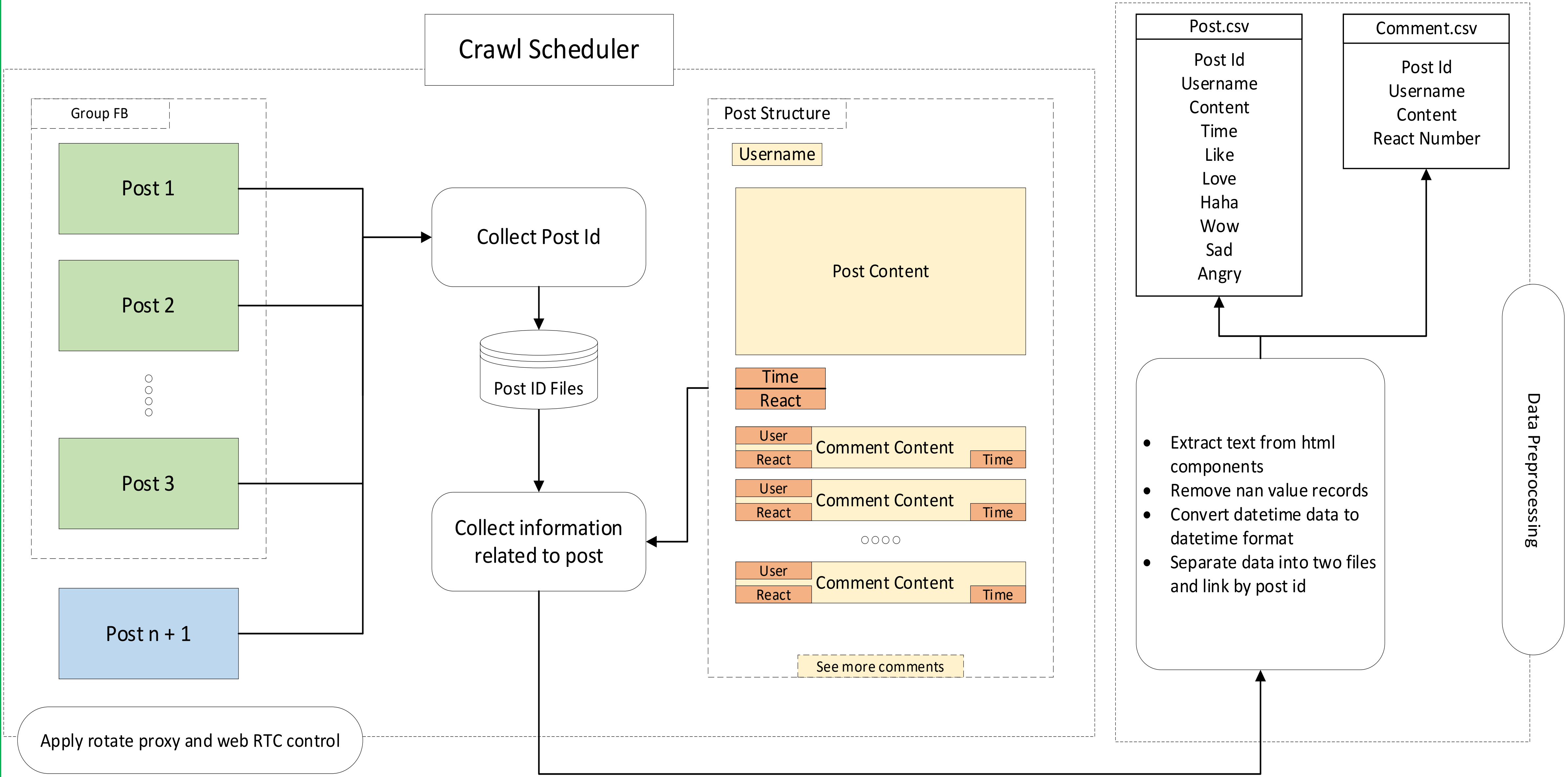


Figure 6. Data Collection

2. Methodology – Data Preprocessing

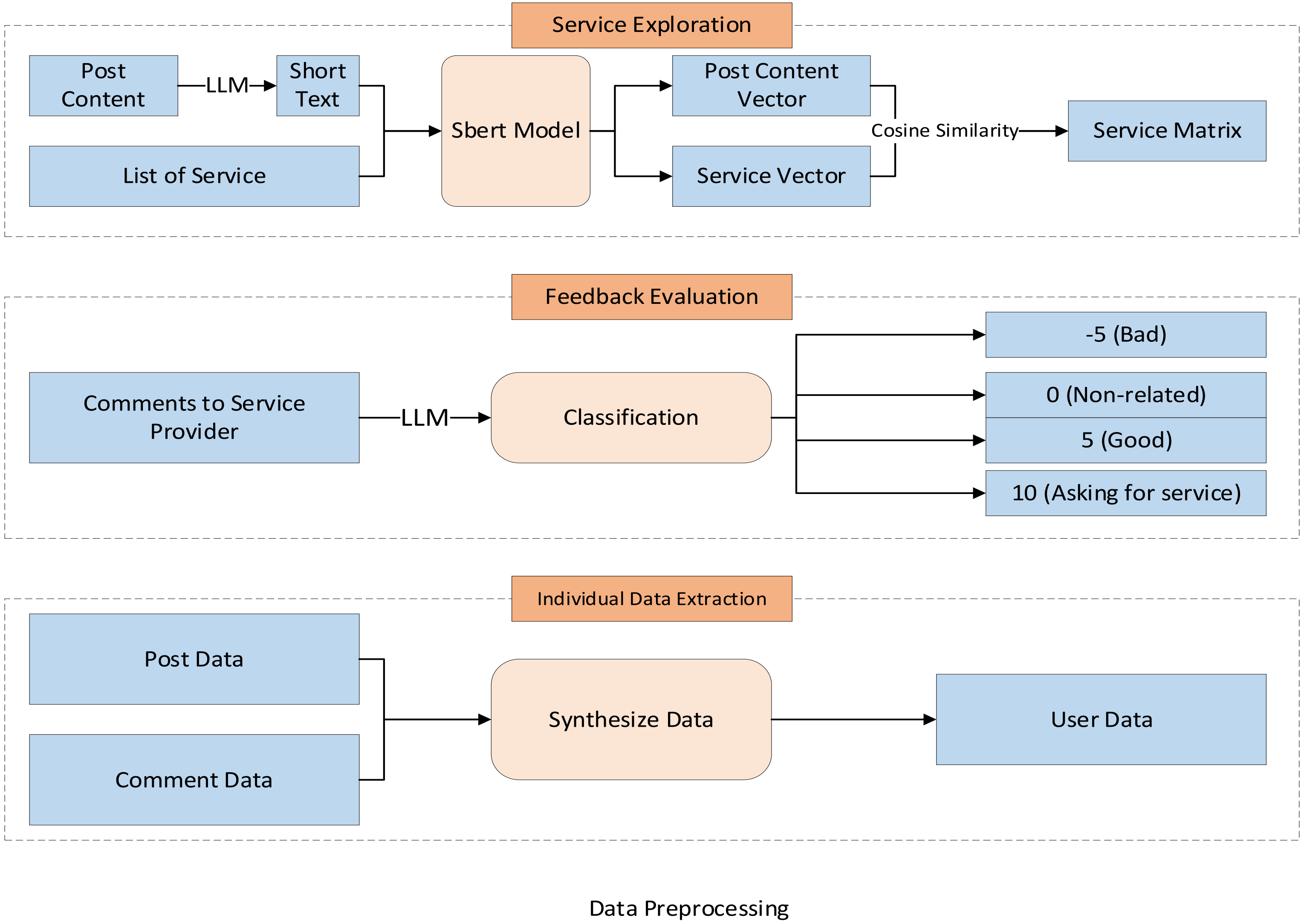


Figure 7. Data Preprocessing

- Ensuring data from module 1 is in a format suitable for analysis, including **cleaning, filtering, and transforming** the data to extract relevant features.
- **Using LLM in Service Exploration and Feedback Evaluation** to quantify the data.

2. Methodology – Data Preprocessing

Few-Shot Prompting	Prompt Chaining
This technique involves providing the LLM with a few examples of the desired output, allowing the model to learn and generalize the task more effectively [3].	Prompt chaining involves breaking down a complex task into a sequence of simpler prompts, which are then executed in a specific order to achieve the desired outcome [4].

- **Prompt techniques integrate into Service Exploration process:**

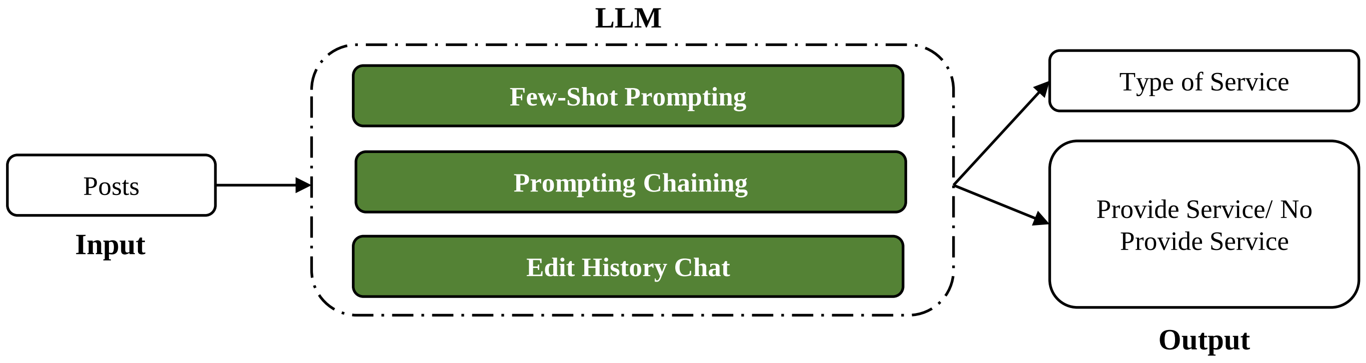


Figure 8. Prompt techniques integrated into Service Exploration process

- **Prompt techniques integrate into Feedback Evaluation process:**

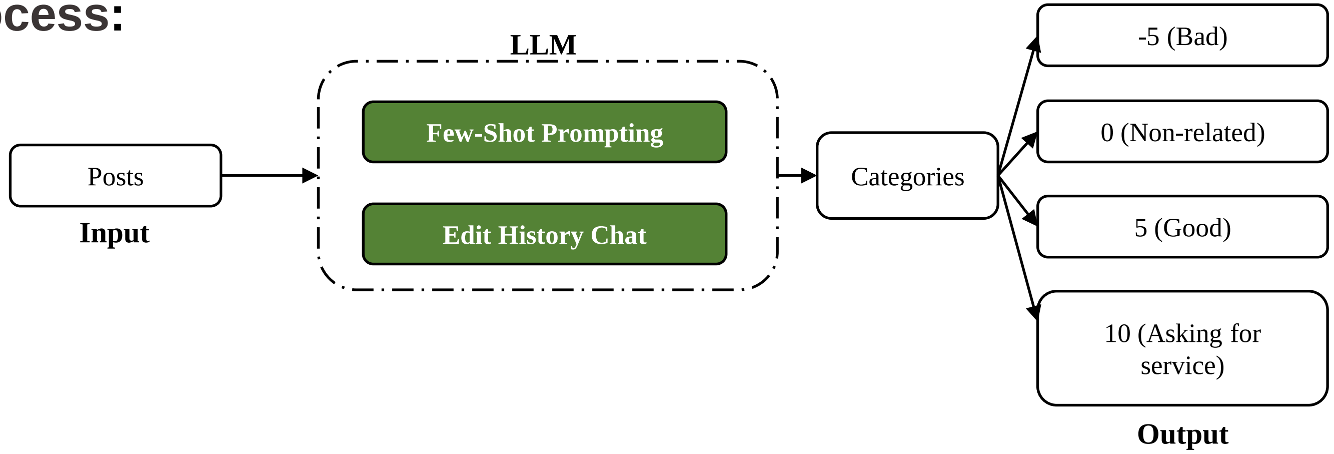


Figure 9. Prompt techniques integrated into Feedback Evaluation process

2. Methodology – Trust Compute & Evaluation

2. Methodology –Trust Compute & Evaluation

Figure 10. Trust Evaluation workflow

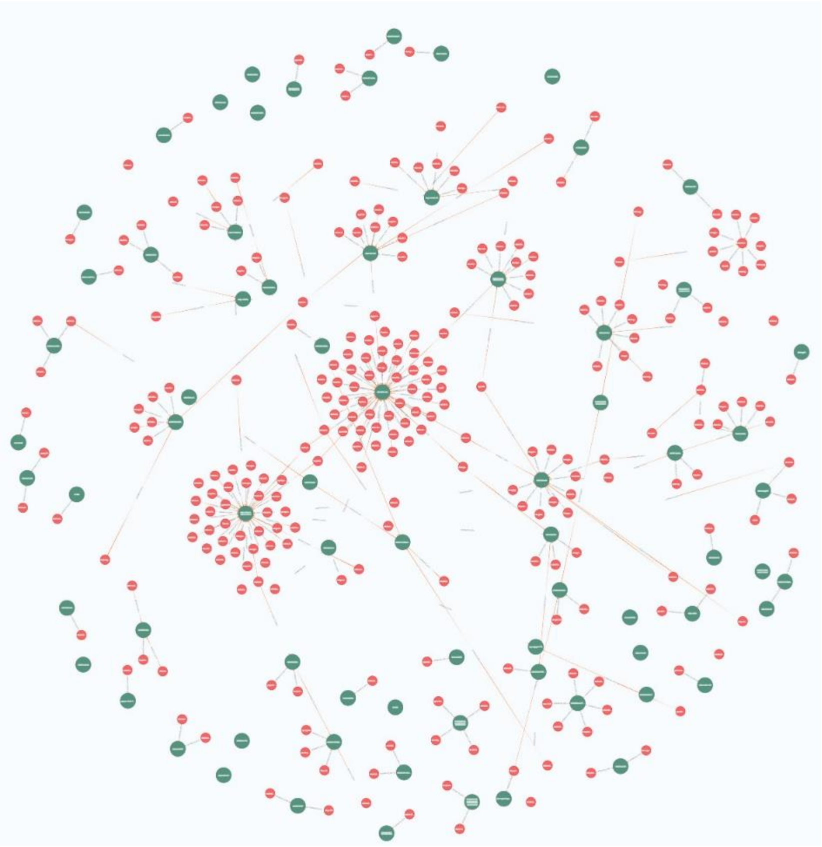


Figure 11. Knowledge Graph after Embedding

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**
        $Strust_{j_{(0)}} \leftarrow (1 - d) + d(\sum_{j \in u_i} \frac{Strust_j}{O_j})$
    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;

Figure 12. TFT Model

2. Methodology – Evaluation Metrics

- **Kendall's Tau:**

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

Where n_+ is the number of concordant pairs; n_- is the number of discordant pairs; N is the total number of pairs elements.

- **Monotonicity:**

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

Where N is the size of the rating vector R; N_r is the number of nodes with the same rank value r .

- **Survey form:** 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

3. Result – Experiment Environment

Fpt University
ĐẠI HỌC FPT

Language Programing: Python 3.8.12

Library Using: selenium, pandas, numpy, networkx, neo4j

Graph Database for KG: Neo4j==5.12.0

Web: Backend java spring boot, front end d3js, bootstrap on jsp

Extension : javascript with chrome API

LLM Model: Vistral 7B Chat model

OS: Windows 11 operating system, Intel Core I7 13650HX CPU and Gefore RTX 4060 8 GB graphics card

3. Experiments & Result - Dataset

GRAPH INFORMATION IN DETAILS

Components	Details
Nodes	1,371 nodes (service providers/ users)
Relationships	1,423 relationships
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

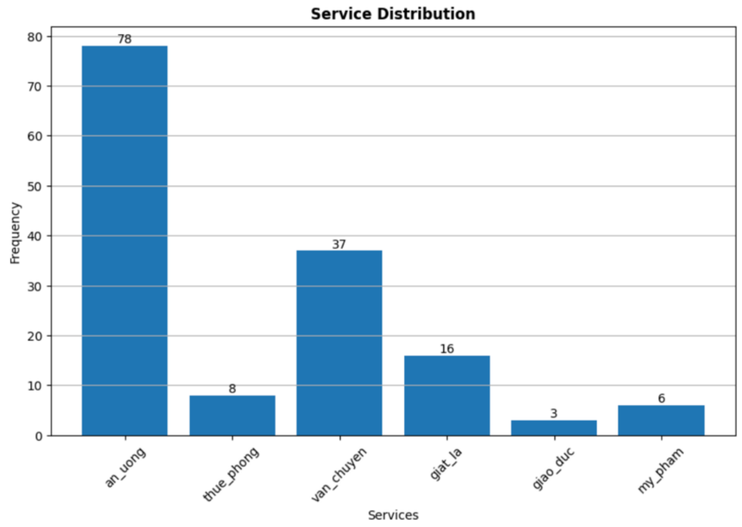


Fig. 13. *Frequency Distribution of Service Categories*

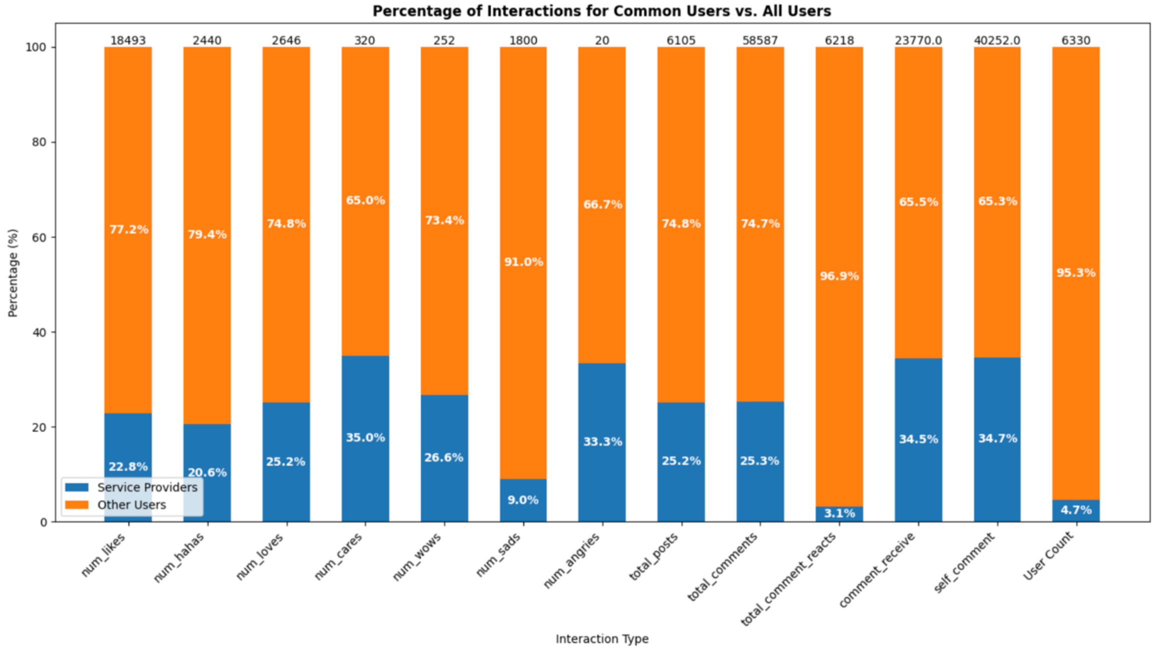


Fig. 14. *Comparative Analysis of Interaction Types: Service Providers vs. Other Users*

3. Result

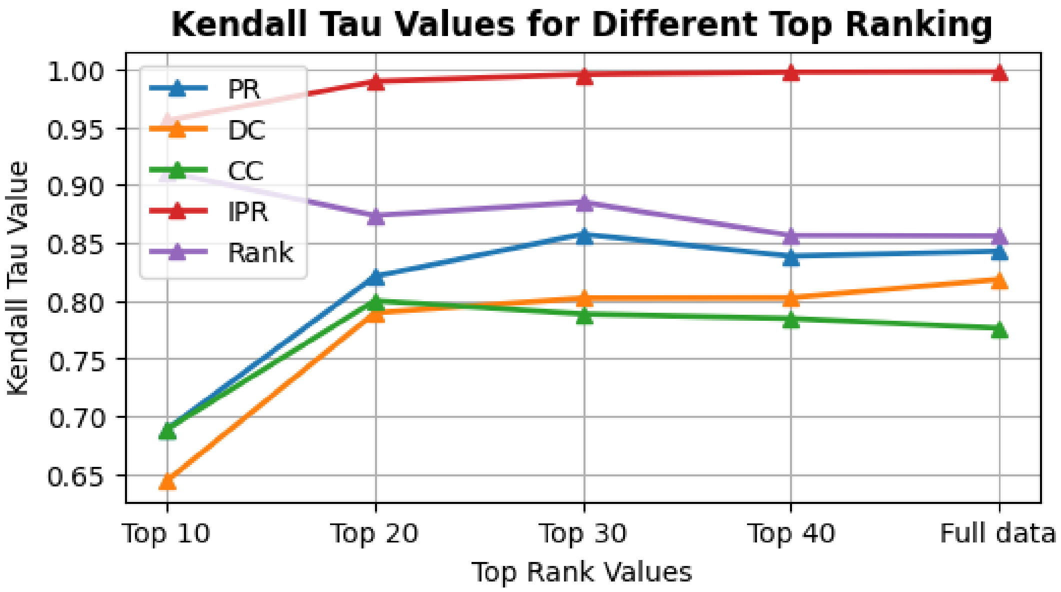


Fig. 15. *Kendall's Tau algorithm for TFTModel value compared to other algorithms (purple-line: Survey Result, blue-line: PageRank, red-line: Improved PageRank, green-line: Closeness Centrality, orange-line: Degree centrality)*

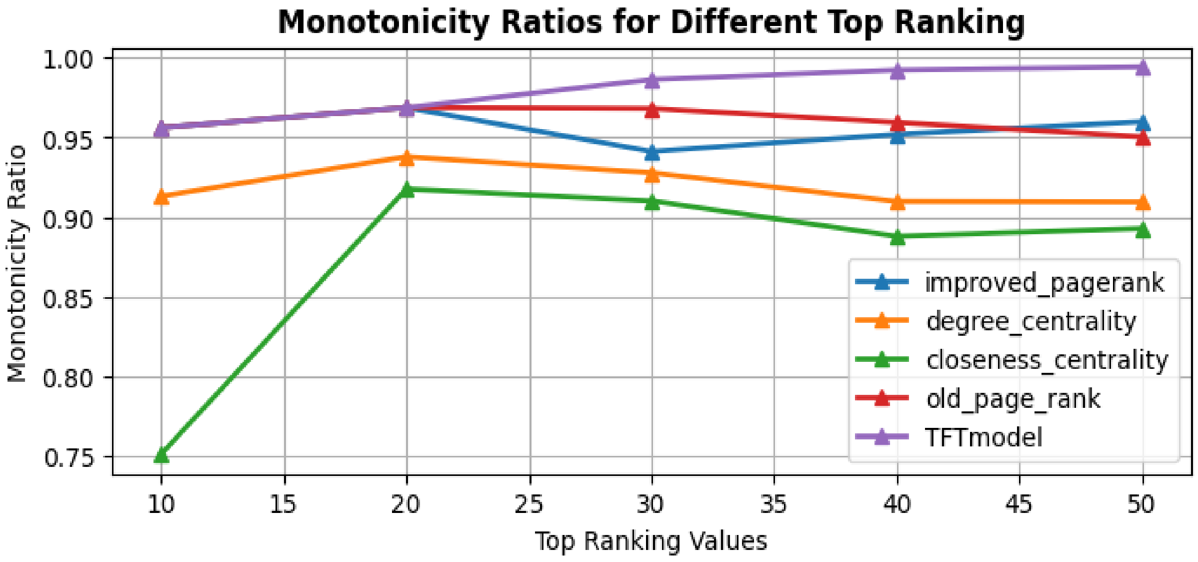


Fig. 16. *Monotonicity algorithm for TFTModel value compared to other algorithms (purple-line: TFTModel, red-line: PageRank, blue-line: Improved PageRank, orange-line: degree centrality: green-line: Closeness Centrality)*

4. Demo

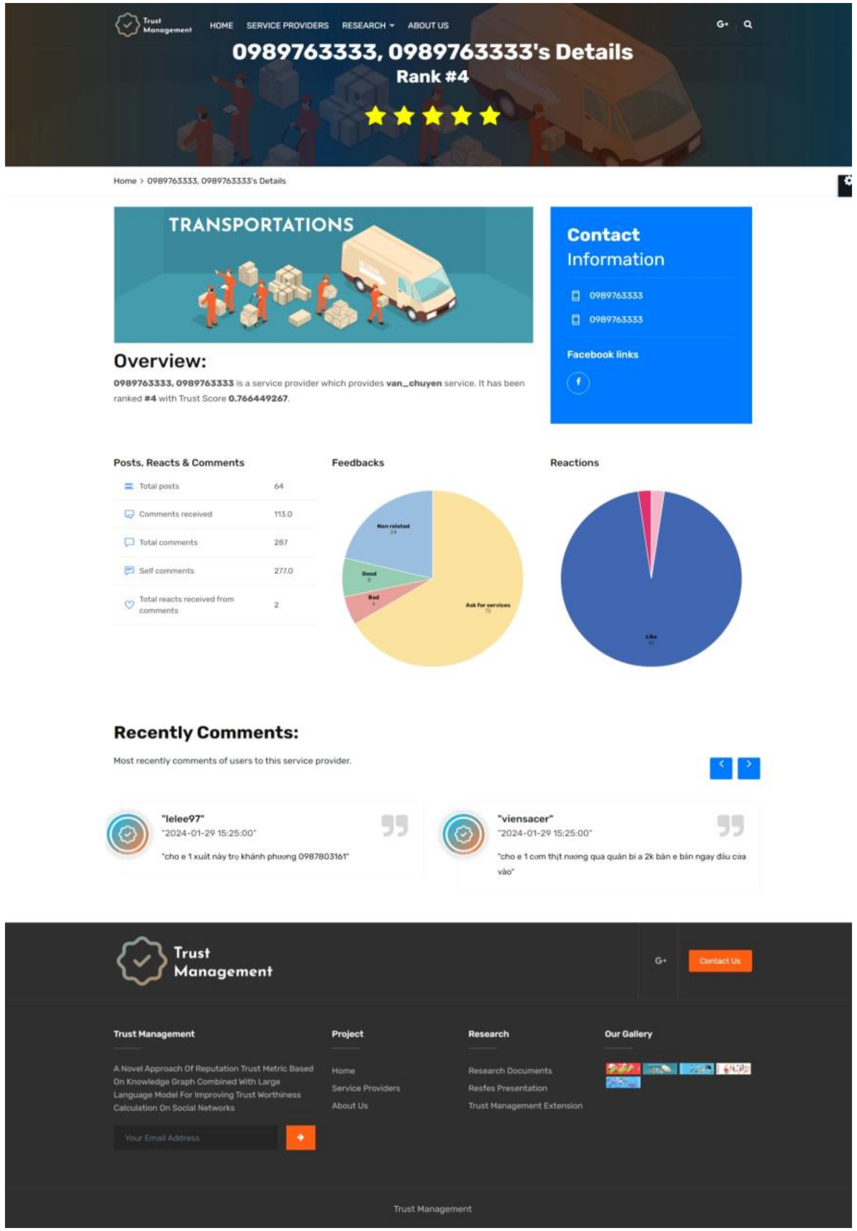


Fig. 17. Website

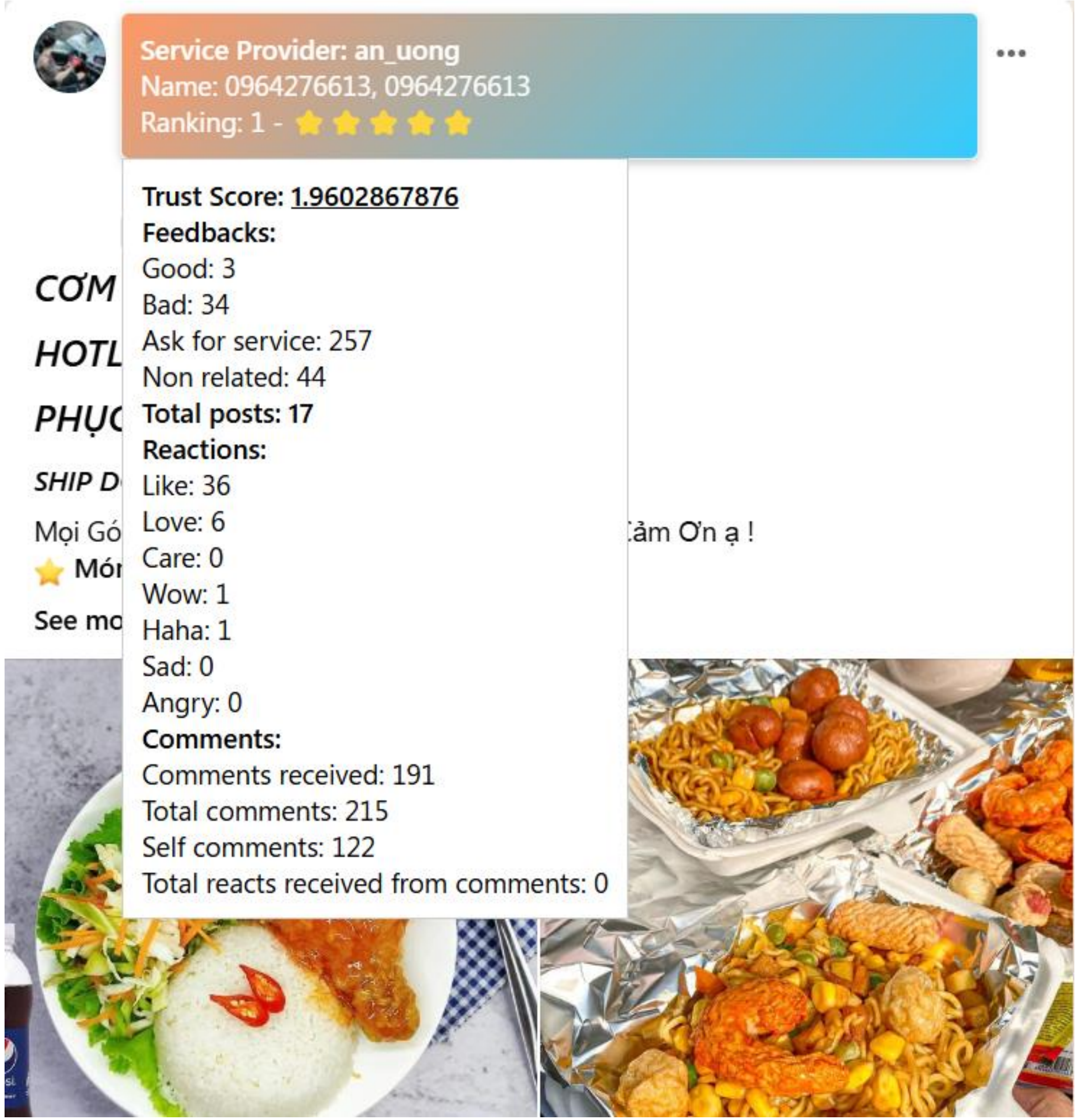
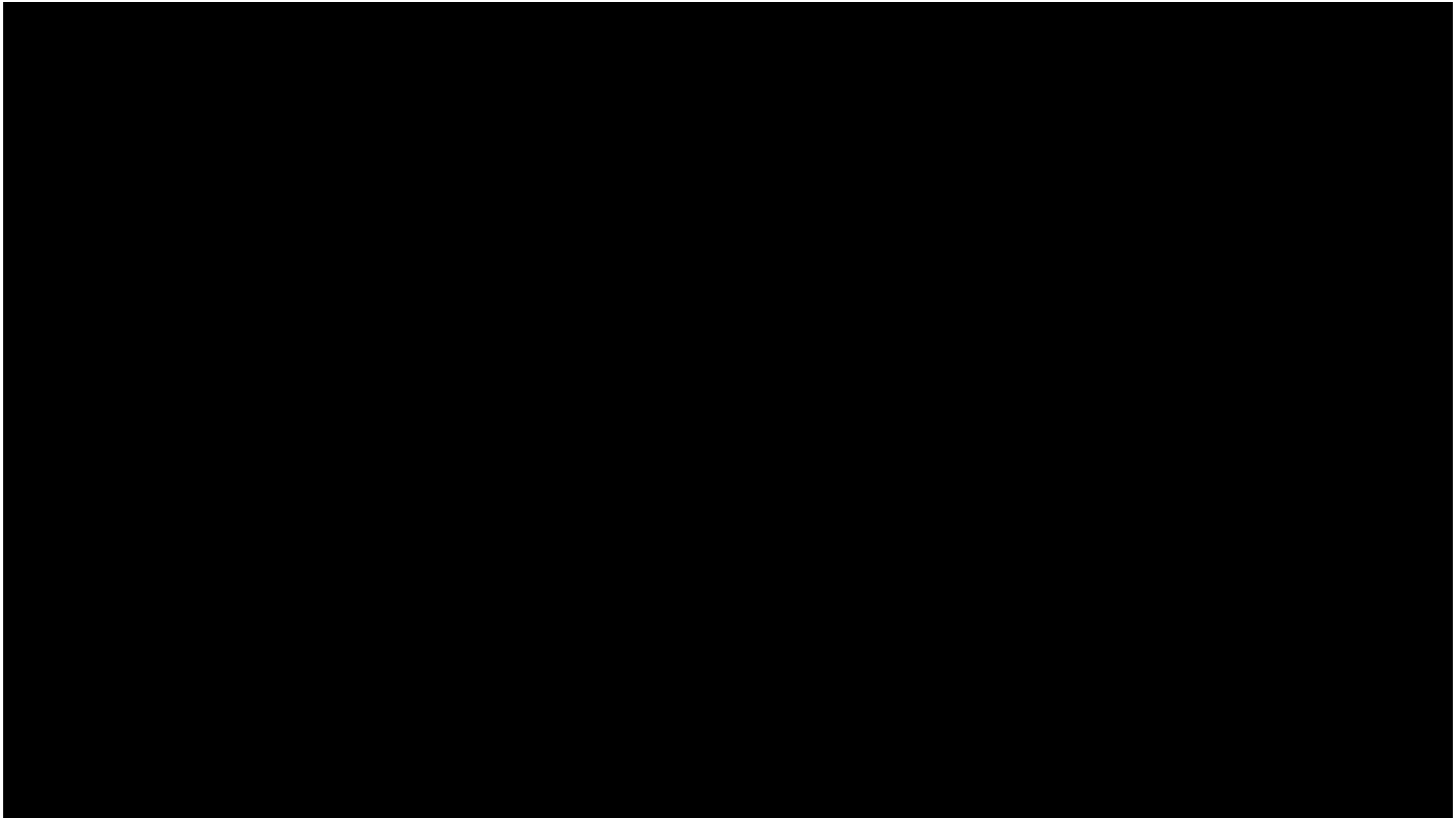


Fig. 18. Extension

4. Demo

4. Demo



Vid. 1. *Website & Extension*

5. References

References

[1] 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.

[2] 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.

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

[4] 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.

[5] 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.

Thank you