Harnessing LLM Conversations for Goal Model Generation from User Reviews

Shuaicai Ren, Hiroyuki Nakagawa and Tatsuhiro Tsuchiya
Graduate School of Information Science and Technology, Osaka University, Suita, Japan

Keywords: LLM, Goal Modeling, User Reviews.

Abstract: User reviews are a valuable resource for developers, as the reviews contain requests for new features and bug reports. By conducting the requirements analysis of user reviews, developers can gain timely insights for the application, which is crucial for continuously enhancing user satisfaction. The goal model is a commonly used model during requirements analysis. Utilizing reviews to generate goal models can assist developers in understanding user requirements comprehensively. However, given the vast number of reviews, manually collecting reviews and creating goal models is a significant challenge. A method for clustering user reviews and automatically generating goal models has been proposed. Nevertheless, the accuracy of the goal models generated by this method is limited. To address these limitations of the existing method and enhance precision of goal model generation, we propose a goal-generation process based on Large Language Models (LLMs). This process does not directly generate goal models from user reviews; instead, it treats goal model generation as a clustering problem, allowing for the visualization of the relationship between reviews and goals. Experiments demonstrate that compared to the existing method, our LLM-based goal model generation process enhance the precision of goal model generation.

1 INTRODUCTION

In modern society, mobile applications (Apps) are playing an increasingly important role in our daily lives. Mobile application platforms, represented by the App Store and Google Play, not only allow users to download apps but also offer a platform for interaction between users and developers. On these platforms, users draft reviews, which include new feature requirements and bug reports (Oriol et al., 2018) (Maalej and Pagano, 2011) (Seyff et al., 2010) (Ma et al., 2015). User reviews are a valuable resource for developers, as reviews offer invaluable insights (Pagano and Maalej, 2013) (Hofmann and Lehner, 2001) (Zowghi, 2018). By conducting the requirements analysis of user reviews, developers deepen their understanding of user requirements, thereby offering version updates that better match user requirements. In the field of requirements analysis, the goal model is one of the most commonly used models, and it can be employed to analyze requirements from user reviews.

The goal model is a basic model in the field of requirements engineering, providing a structured framework to describe what functions a system needs and how to implement these functions. Within the goal model, goals are arranged in a hierarchical structure, where the root goal is refined into sub-goals, ultimately forming a comprehensive goal structure. This layered structure helps in deeply understanding the interdependencies among goals. The goal model contains multiple elements, such as conflicts and soft goals. Conflicts refer to situations where achieving one goal may obstruct the realization of another, while soft goals aim to capture non-functional requirements. The primary advantage of the goal model is that it allows developers to define and understand requirements with clarity. When goals conflict, the goal model can support crucial decision-making. Compared to directly analyzing user reviews, making goal models ensures that user reviews match the app's goals. By connecting reviews with goals, it is easier to figure out what goals users care about and if the new features they want might conflict with current goals. This helps developers know which user requirements are most urgent and helps them make better updates.

While there are numerous advantages to employing a goal model for the analysis of user reviews, the manual construction of a goal model presents a significant challenge. This challenge mainly arises from the

Ren, S., Nakagawa, H. and Tsuchiya, T.
Harnessing LLM Conversations for Goal Model Generation from User Reviews.
DOI: 10.5220/0012352200003636

In *Proceedings of the 16th International Conference on Agents and Artificial Intelligence (ICAART 2024) - Volume 3*, pages 385-392
ISBN: 978-989-758-680-4; ISSN: 2184-433X

vast number of reviews and only a small proportion of them contain requirements or bug reports (Licorish et al., 2015) (Pagano and Maalej, 2013) (Chen et al., 2014). Consequently, the process of manually reading and summarizing reviews becomes a labor-intensive and time-consuming task. To automate the process of utilizing user reviews for goal model generation, we proposed a method for clustering reviews (Ren et al., 2022). This clustering method consists of two components: the Latent Dirichlet Allocation (LDA) topic model (Blei et al., 2003) and a distance-based clustering algorithm. The method defines the root goal as containing all user reviews, and the topics generated by the LDA topic model are treated as sub-goals under the root goal. To refine sub-goals from the generated topics, a distance-based clustering algorithm is introduced. This method simplifies the process of generating goal models from user reviews, significantly reducing the required manpower. However, it must be acknowledged that this method has certain limitations. For instance, its accuracy is suboptimal, and the generated goals do not have a one-to-one correspondence with the requirements. To enhance the precision of goal model generation from user reviews, we explore the potential of leveraging large language models (LLMs) for goal model generation. By harnessing the capabilities of LLMs, our objective is to enhance the precision and efficacy of the goal modeling process, providing a more accurate representation of user requirements and preferences.

This technology holds the potential to make contributions to agents. The application of LLMs in goal model generation has several impacts. LLMs help agents better understand and pull key information from a wide range of user reviews. These reviews, coming from many users, provide valuable information and preferences that, when processed effectively, can significantly contribute to agent development. By grouping and refining user reviews, agents can get a deeper understanding of what users really want and their main concerns. This deeper understanding allows them to give more relevant and personalized responses, making the user experience better.

LLMs represent an advanced class of natural language processing (NLP) models, notable for their extensive size and parametric complexity. In this research, we harnessed GPT-4 (OpenAI, 2023), recognized as a state-of-the-art LLM, for the purpose of clustering user reviews and generating goal models. The goal model generated by GPT-4 has higher accuracy than the goal model generated by the existing method. However, using GPT-4 to directly generate goal models comes with certain drawbacks, such as producing repetitive or incorrect goals and failing to describe the relationships between generated goals and reviews. To address these issues, we propose a novel goal model generation process. This process does not involve the direct use of GPT-4 to generate goal models; instead, it begins with clustering and analyzing user reviews before generating the goal models. The experimental results demonstrate that, compared to the existing method, the use of the proposed process with the GPT-4 method gains higher-precision goal models. This not only saves time but also enhances developers' understanding of the importance of each goal.

The contributions of this study are as follows: First, we introduce a goal model generation method based on LLMs, offering an alternative method for the automation of goal model generation. Second, the proposed method treats goal model generation as a clustering problem, allowing developers to understand the relationship between user reviews and goals. Third, compared to the existing method, the proposed method enhances the precision of goal model generation.

The following sections of this paper are organized as: Section 2 introduces related work relevant to our research. Section 3 presents the existing method for goal model generation. Section 4 introduces GPT-4 and the proposed process for generating goal models utilizing GPT-4. Section 5 showcases comparative experiments between different goal model generation methods. Section 6 evaluates the proposed goal model generation process. Finally, in Section 7, we summarize this study and outline future work.

2 RELATED STUDIES

In a recent development, Jiang et al. (Jiang et al., 2019) introduced SAFER, a novel approach that enables the automatic extraction of features from application descriptions and the identification of analogous applications based on API names and extracted features. SAFER further undertakes the aggregation and recommendation of features from identified analogous applications. On a related note, Dkabrowski et al. (Dąbrowski et al., 2023) conducted empirical research into three distinct opinion mining methods: GuMa (Guzman and Maalej, 2014), SAFE (Johann et al., 2017), and ReUS (Dragoni et al., 2019). These methods underwent evaluation through two distinct tasks encompassing feature extraction and sentiment analysis. The research outcomes proffered valuable insights by suggesting that the efficacy of these methods might be lower than originally reported. Moreover, Malik et al. (Malik et al., 2020) proposed a comprehensive approach for the extraction of opinions from user reviews. Their approach is particularly geared towards assisting developers and users in the automated extraction and comparison of features across a spectrum of mobile applications. It is important to emphasize that the aforementioned studies primarily pivot around the domain of review analysis, and their primary focus does not lie in goal model identification.

It is important to note that several researchers believe that LLMs have the potential to revolutionize existing software development processes. This has led to the proposal of numerous methods that leverage LLMs for software modeling. Nakagawa et al. (Nakagawa and Honiden, 2023) introduced a semi-automated process for goal model generation that employs generative AI founded on the MAPE-K loop mechanism. Their two case studies demonstrate that this process, based on the MAPE-K loop mechanism, is efficacious in goal model construction without omitting any goal descriptions. Additionally, Cámara et al. (Cámara et al., 2023) conducted a comprehensive investigation into GPT-4's performance in modeling tasks and its utility to modeling personnel, while simultaneously identifying its principal limitations. Their research findings underscore that the current iteration of GPT-4 exhibits limited efficacy in software modeling, especially when compared to its capabilities in code generation. It exhibits variegated syntax and semantic defects, lacks response consistency, and faces scalability challenges. Ding et al. (Ding and Ito, 2023) introduced the 'Self-Agreement' framework, aimed at autonomously seeking consensus among diverse opinions using data generated by large language models (LLMs), without the need for extensive manual annotation. They utilized GPT-3 to generate multiple opinions for each question in a question dataset and subsequently employed a BERT model to evaluate the consistency of each opinion, selecting the most consistent one. Their research focused on finding consensus among diverse opinions, whereas our method centers on analyzing the consistency of reviews and generating goal models.

Chen et al. (Chen et al., 2023) reported the preliminary experimental results of goal model generation using GPT-4. They first explored GPT-4's understanding of the Goal-oriented Requirement Language (GRL) and then employed four prompt combinations to guide the generation of GRL models in two case studies. One case was a well-documented topic in the goal modeling domain, while the other was the opposite. The experimental results indicate that GPT-4 possesses extensive knowledge related to goal models and that the generated goal models are valuable. Notably, all three methods employ LLMs to generate goal models. However, it is crucial to point out that, unlike our method, they do not leverage user reviews in the goal model generation process.

3 EXISTING METHOD

We proposed a method for creating a goal model by clustering user reviews (Ren et al., 2022). This method involves two key components: the LDA topic model and a distance-based clustering algorithm. The LDA topic model is responsible for generating goals from all reviews, while the distance-based clustering algorithm refines these goals. The LDA model is a widely utilized probabilistic topic modeling technique for the analysis of extensive unstructured textual data in academic research (Papadimitriou et al., 2000) (Blei et al., 2003). The goal representing all reviews is regarded as the root goal, while the topics generated by the LDA model are considered as sub-goals of the root goal. While it's possible to refine and create the goal model further by applying LDA modeling to reviews within each topic, this method may not be reliable when dealing with a limited number of reviews (Hajjem and Latiri, 2017).

Given the limitations of LDA topic modeling in such scenarios, a distance-based clustering algorithm is proposed to facilitate further refinement. For each topic, the reviews are vectorized, and Ward's method is employed to calculate distances between vectors, resulting in the creation of compact, evenly sized clusters (Szmrecsanyi, 2012). These clusters are visually represented by a dendrogram. Clusters with similar distance values are assigned as sub-goals under the same parent goal. This method follows a top-down approach, creating boundary lines. Clusters above the boundary line become parent goals, while those below it become sub-goals. The boundary line's value is determined by the cluster distances and manually selected parameters. This automated method of generating goal models from user reviews aids in a deeper understanding of user requirements. By combining LDA and the distance-based clustering algorithm, this clustering method addresses the challenges of analyzing numerous reviews and automatically identifies main topics and their hierarchical relationships, simplifying the goal model generation process. Nevertheless, this method still faces specific limitations, primarily accuracy issues. Current methods employ Ward's method for review clustering, where reviews are first converted into vectors, and review similarity is determined based on vector distance. Neverthe-

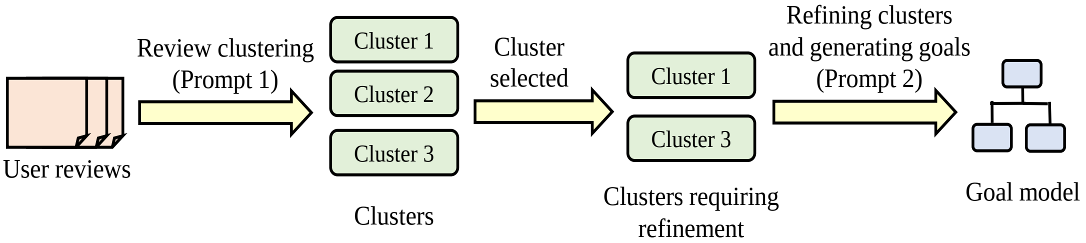
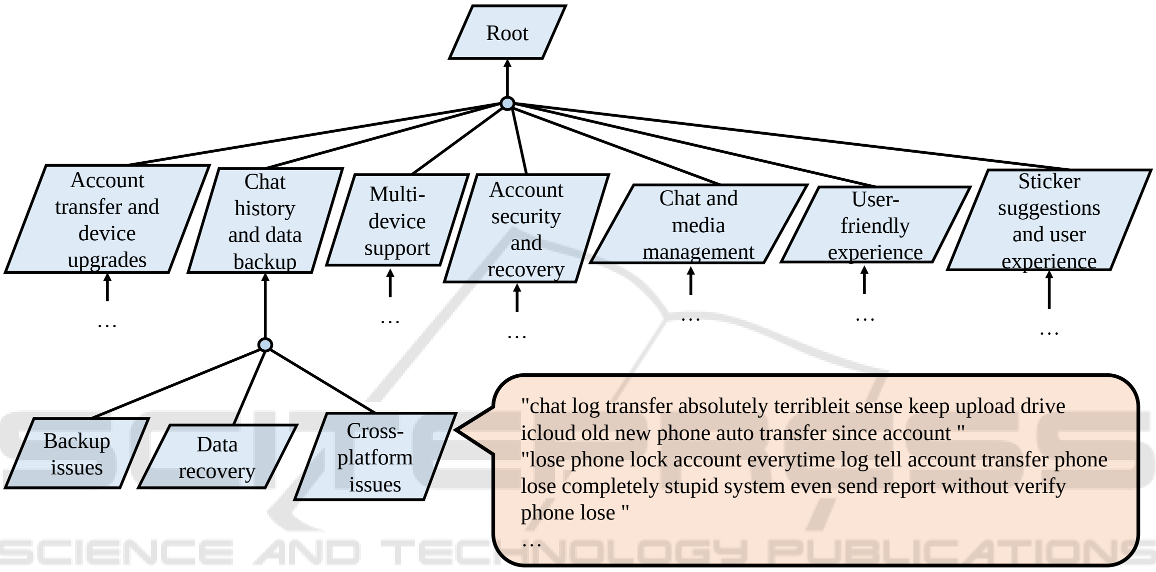


Figure 1: Overview of the proposed process for generating goal models using GPT-4.


Root
Account transfer and device upgrades
Chat history and data backup
Multi-device support
Account security and recovery
Chat and media management
User-friendly experience
Sticker suggestions and user experience
…
Backup issues
Data recovery
Cross-platform issues

"chat log transfer absolutely terribleit sense keep upload drive icloud old new phone auto transfer since account "
"lose phone lock account everytime log tell account transfer phone lose completely stupid system even send report without verify phone lose "
…

Figure 2: Goal model generated by the proposed process.

less, research indicates that vector similarity does not always reflect the similarity of the requirements described in the reviews (Devine et al., 2022). Within the same cluster, reviews may share common aspects, but these shared aspects do not necessarily indicate identical requirements.

To enhance goal model generation precision and improve developers' understanding of the generated goal model, we explore the possibility of utilizing GPT-4 to cluster reviews and generate goal models.

4 LLM-BASED GOAL MODEL GENERATION METHOD

GPT-4, which stands for "Generative Pre-trained Transformer 4," is a state-of-the-art language model developed by OpenAI. GPT-4's primary objective is to facilitate interactive conversations with users, offering responses that are contextually coherent across a wide spectrum of prompts and inquiries. While there are many advantages to generating goal models directly using GPT-4, several challenges also exist. For example, the relationship between user reviews and the goals cannot be visualized. Sometimes, the information developers obtain from the goals is insufficient to fully understand user requirements. In cases like this, if the goal model is generated using review clustering methods, developers can check reviews related to the goals to gather additional insights. However, in goal models generated using GPT-4, reviews related to the goals are not presented comprehensively. Even if you ask GPT-4 to display all the reviews, it provides only a limited set of reviews, making it challenging for developers to understand user requirements. Additionally, the prioritization of goals is based on the frequency of corresponding requirements mentioned in the reviews. The lack of visualization for the relationship between goals and the associated reviews decreases the credibility of goal prioritization.

To address the issue of the lack of visualized relationships between goals and reviews in the goal model directly generated using GPT-4, we propose a novel process for generating goal models using GPT-4. This process frames the task of generating goal models as a clustering problem, where clusters represent the goals. GPT-4 provides the cluster to which each review belongs, thus visualizing the relationship between goals and reviews. Subsequently, developers can refine some or all of the goals based on factors such as the number of reviews or the content of reviews. Figure 1 shows the overview of the proposed process.

This process consists of two steps. The first step involves clustering reviews using GPT-4, in this step, the generated clusters are regarded as goals. The prompt for this step are as follows:

> Prompt 1: Can you cluster the following reviews?

The second step occurs after developers analyze the generated goals and decide which goals to refine. GPT-4 is then employed to refine the selected goals, with prompts for this step as follows:

> Prompt 2: Can the first category be refined, and if so, what would the relevant reviews in the subdivided categories look like? By "relevant reviews," I mean the reviews I provided earlier. There is no need to generate relevant reviews; all reviews should belong to the first category classified earlier. Each comment should belong to only one subcategory, and each subcategory should be akin to a goal in the goal model in the requirement model.

It is crucial to include the statement "Cannot generate reviews" in the prompt for the second step to prevent generation errors by GPT-4. Additionally, it should be specified that each review can belong to only one goal to ensure that redundant reviews do not influence developers' assessments of goal importance. By adopting the proposed process, developers can maintain the advantages of using GPT-4 to generate goal models while visualizing the relationships between reviews and goals. Furthermore, it provides a more detailed and time-efficient approach for developers to analyze and refine goals.

In summary, utilizing the proposed process for generating goal models from user reviews offers numerous benefits, including high goal coverage, selective refinement of goals, and streamlined data processing. These advantages position the proposed process as a valuable tool for extracting and comprehending user requirements, facilitating effective decision-making, and enhancing the development of user-centric applications.

5 EXPERIMENT

The purpose of the experiment is to assess which of the two, the existing method and the GPT-4 method with the proposed process, is more similar to the manually created goal model, including both structural and content similarities. We have collected 150 user reviews from the App Store. These reviews are from Line, Google Docs, and YouTube, with each app contributing 50 reviews. For each set of user reviews for these apps, we used three different methods for goal model generation: the existing method, the GPT-4 method with the proposed process, and the manual method. Figure 2 illustrates a portion of the goal model generated using the GPT-4 with the proposed process, with user reviews sourced from Line. To create the manual model, we manually examined each review and determined which goal it should belong to.

We initially focus on evaluating the structure of the generated goal models. Since goal models have a tree-like structure, we utilize Tree-Edit-Distance-based Similarity (TEDS) (Zhong et al., 2020) to assess the similarity between the models generated by the two methods and the manually created model. TEDS is a normalized variant of the Tree-Edit-Distance (TED), and its calculation is as follows:

$$\mathit{TEDS(G,G\_m)} = 100 - \left(\frac{\mathit{EditDist(G,G\_m))}}{\mathit{max(|G|,|G\_m|)}} \times 100\right), \quad (1)$$

where *EditDist(G,G_m)* is computed as the minimum number of operations, comprising both *Move* and *Join* operations, necessary to transform the generated goal model *G* into the goal model *G_m*, which is manually created. The value of *EditDist(G,G_m)* is determined through manual computation. *max(|G|,|G_m|)* represents the maximum number of goals present within goal models *G* and *G_m*. Consequently, the higher the degree of similarity between the goal model *G* and the goal model *G_m*, the larger the resulting *TEDS* value. In the case of complete equivalence between *G* and *G_m*, *TEDS* returns a value of 100. Table 1 demonstrates the TEDS values for three apps. The average TEDS value of the GPT-4 method is 26 points higher than the existing method, indicating that the goal model generated by GPT-4 is more similar to the manually created goal model.

In terms of evaluating the content of the generated goals, we employ precision and recall to assess both the existing method and the GPT-4 method. For

Table 1: Results of TEDS.

	Line	YouTube	Google Docs	Avg
Existing method	70	56	67	62
GPT-4 with proposed process	100	89	75	88

a goal *A* generated by the existing method or the GPT-4 method, we found a goal *A'* identified by the manual goal model that allows the best precision to be achieved for the goal *A*. Table 2 lists the precision and recall for each goal.

6 DISCUSSION

From Table 2, we observe that the goal model generated by the GPT-4 method exhibits higher precision and recall. In contrast to the LDA topic model, GPT-4 does not require the prior specification of the number of topics and eliminates the need for extensive preprocessing steps. Even when dealing with a limited number of reviews, the GPT-4 method does not experience a decline in accuracy. For the LDA method, having too many uninformative reviews or too few reviews can reduce the clustering accuracy. Another component of the existing method is the distance-based clustering method, where distances are computed using Ward's method. This method involves first transforming reviews into vectors and subsequently determining the similarity between reviews based on the vector distances. Nevertheless, studies have demonstrated that the similarity of vectors may not always correspond to the likeness of requirements articulated within the reviews (Devine et al., 2022). While reviews grouped within the same cluster may exhibit shared elements, the presence of these commonalities does not invariably signify congruent requirements. Owing to its robust performance, GPT-4 can clearly understand the requirements described in user reviews. GPT-4 underwent extensive pre-training on a vast corpus of textual data. As a result, GPT-4 can comprehend intricate content within user reviews, which often contain colloquialisms, domain-specific jargon, and various forms of expression. Even if user reviews are incomplete, use slang, or unconventional punctuation, GPT-4 can filter potential intent and information from the noise.

By comparing the direct generation of goal models using GPT-4 with the goal models generated through our proposed process, we identify several advantages of the proposed process, which include:

- By clustering reviews, the workload of GPT-4's generation process can be significantly reduced. Developers have the option to refine only the most critical goals, instead of diving into the fine details of every single objective. This method becomes especially advantageous when dealing with a large volume of user reviews, as clustering aids in efficiently managing and analyzing extensive datasets, thereby reducing the complexity associated with processing substantial amounts of data. For developers dealing with significant quantities of user-generated content, such as product or service reviews, this reduction in complexity is invaluable. It not only accelerates the goal model generation process but also aids in maintaining the quality of the analysis. As a result, developers can gain meaningful insights from a vast dataset without being overwhelmed by its scale.
- The proposed process can enhance the consistency of the generated goal models. Clustering can aid in ensuring that the generated goal model remains consistent within similar clusters of reviews, thereby enhancing user experience and comprehension. A noteworthy advantage of this clustering process is its ability to scrutinize the generated goals alongside their related reviews. This critical examination phase serves as a quality control mechanism, allowing for the identification and correction of ambiguous or erroneous goals. Such corrections significantly reduce the likelihood of potential errors during the goal model generation phase.
- Clustering can serve as an intermediary step, greatly facilitating developers in iteratively improving the generated goal model. Based on the outcomes of clustering, developers can fine-tune and optimize the generated goal model progressively to enhance its quality. By examining the clustered goals and the feedback derived from these clusters, developers can make data-driven decisions to prioritize certain requirements over others. This iterative approach fosters a responsive development environment where the generated goal model evolves alongside user feedback. As a result, the goal model becomes increasingly aligned with the users' expectations, ensuring that the final product or service is more user-centric and attuned to their needs.

Table 2: Precision and recall of generated goals. E_Goals are generated by the existing method, and G_Goals are generated by GPT-4 with the proposed process. Avg means the average value of the precision or recall for generated goals.

App	Line								
Goal	E_Goal 1	E_Goal 2	E_Goal 3	E_Goal 4	E_Goal 5	E_Goal 6	E_Goal 7	E_Goal 8	**E_Avg**
Precision	0.56	1	0.33	0.67	0.5	0.2	0.5	0.25	**0.5**
Recall	0.45	0.89	0.2	0.25	0.57	0.33	0.17	0.67	**0.44**
Goal	G_Goal 1	G_Goal 2	G_Goal 3	G_Goal 4	G_Goal 5	G_Goal 6			**G_Avg**
Precision	0.8	1	0.67	0.83	0.5	0.88			**0.78**
Recall	1	0.9	0.75	0.86	0.33	0.71			**0.76**

App	YouTube							
Goal	E_Goal 1	E_Goal 2	E_Goal 3	E_Goal 4	E_Goal 5			**E_Avg**
Precision	0.5	0.5	0.33	0.5	0.6			**0.49**
Recall	0.25	0.33	0.17	0.67	0.43			**0.37**
Goal	G_Goal 1	G_Goal 2	G_Goal 3	G_Goal 4	G_Goal 5	G_Goal 6	G_Goal 7	**G_Avg**
Precision	0.8	0.67	0.67	0.5	0.8	0.67	0.65	**0.68**
Recall	0.75	1	0.5	0.85	0.86	0.75	0.45	**0.74**

App	GoogleDocs								
Goal	E_Goal 1	E_Goal 2	E_Goal 3	E_Goal 4	E_Goal 5	E_Goal 6	E_Goal 7	E_Goal 8	**E_Avg**
Precision	0.43	1	0.5	0.42	0.67	0.25	0.5	0.4	**0.52**
Recall	0.75	0.66	0.18	0.6	0.4	0.13	0.33	0.25	**0.41**
Goal	G_Goal 1	G_Goal 2	G_Goal 3	G_Goal 4	G_Goal 5				**G_Avg**
Precision	0.71	0.57	0.6	0.75	1				**0.73**
Recall	0.63	0.67	0.43	0.71	0.2				**0.53**

7 CONCLUSIONS

In this study, we explored the potential of utilizing GPT-4 to generate goal models and proposed a novel goal model generation process. To improve the generation accuracy of the goal model, we employed GPT-4 to generate goal models. However, a limitation of the method that directly generates goal models from user reviews is that the relationship between goals and reviews is not visualized. To address this limitation, we introduced a new process that treats goal model generation as a clustering problem. This process significantly saves developers' time and enhances their understanding of the goal content. Experimental results indicate that the accuracy of generating goal models using the proposed process is higher than that of the existing method. Regarding future research directions, we have outlined the following objectives:

Enhancing stability in goal model generation: When utilizing GPT-4 for the classification of user reviews, it is important to consider that the outcomes generated may not always be consistent. Specifically, the results might exhibit variations, such as a classification not based on requirements but rather influenced by emotional content. This phenomenon highlights an essential aspect of working with AI-based language models, where several factors contribute to the unpredictability of the results. GPT-4 does not possess the ability to discern the "correct" method of classification a priori. Its responses are determined by patterns and information gleaned from its training data. As a result, the quality of the responses depends on the quality and specificity of the training data and the formulation of the user's query. To enhance the stability and reliability of user review classification using GPT-4, strategies such as fine-tuning the model on domain-specific data, providing clear instructions to the model, or post-processing its outputs may be considered. These strategies can help align the model's responses more closely with the specific goals of the task, reducing the variability in outcomes, and minimizing the impact of emotional content on classification results.

Time complexity reduction: Although GPT-4 exhibits the capability to generate goal models from user reviews, it is essential to acknowledge the substantial time investments linked to the current implementation of this process. Even when dealing with a relatively modest review dataset, comprising fewer than 100 reviews, a noteworthy amount of time and computational resources is imperative. This temporal overhead may potentially impede the practical applicability of the approach. In response to this concern, our future research initiatives are dedicated to the refinement of GPT-4's capabilities, encompassing strategies such as fine-tuning and few-shot learning. Our ultimate goal is the reduction of time complexity while upholding the precision of the generated goal models.

REFERENCES

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Cámara, J., Troya, J., Burgueño, L., and Vallecillo, A. (2023). On the assessment of generative ai in modeling tasks: an experience report with chatgpt and uml. *Software and Systems Modeling*, pages 1–13.

Chen, B., Chen, K., Hassani, S., Yang, Y., Amyot, D., Lessard, L., Mussbacher, G., Sabetzadeh, M., and Varró, D. (2023). On the use of GPT-4 for creating goal models: An exploratory study. In *2023 IEEE 31st International Requirements Engineering Conference Workshops (REW)*, pages 262–271. IEEE.

Chen, N., Lin, J., Hoi, S. C., Xiao, X., and Zhang, B. (2014). AR-miner: mining informative reviews for developers from mobile app marketplace. In *Proc. of the 36th International Conference on Software Engineering*, pages 767–778. ACM.

Dąbrowski, J., Letier, E., Perini, A., and Susi, A. (2023). Mining and searching app reviews for requirements engineering: Evaluation and replication studies. *Information Systems*, page 102181.

Devine, P., Tizard, J., Wang, H., Koh, Y. S., and Blincoe, K. (2022). What's inside a cluster of software user feedback: A study of characterisation methods. In *2022 IEEE 30th International Requirements Engineering Conference (RE)*, pages 189–200. IEEE.

Ding, S. and Ito, T. (2023). Self-agreement: A framework for fine-tuning language models to find agreement among diverse opinions. *arXiv preprint arXiv:2305.11460.*

Dragoni, M., Federici, M., and Rexha, A. (2019). An unsupervised aspect extraction strategy for monitoring real-time reviews stream. *Information processing & management*, 56(3):1103–1118.

Guzman, E. and Maalej, W. (2014). How do users like this feature? a fine grained sentiment analysis of app reviews. In *2014 IEEE 22nd international requirements engineering conference (RE)*, pages 153–162.

Hajjem, M. and Latiri, C. (2017). Combining ir and lda topic modeling for filtering microblogs. *Procedia Computer Science*, 112:761–770.

Hofmann, H. F. and Lehner, F. (2001). Requirements engineering as a success factor in software projects. *IEEE software*, 18(4):58–66.

Jiang, H., Zhang, J., Li, X., Ren, Z., Lo, D., Wu, X., and Luo, Z. (2019). Recommending new features from mobile app descriptions. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 28(4):1–29.

Johann, T., Stanik, C., Maalej, W., et al. (2017). SAFE: A simple approach for feature extraction from app descriptions and app reviews. In *Proc. of the 2017 IEEE 25th international requirements engineering conference (RE)*, pages 21–30. IEEE.

Licorish, S. A., Tahir, A., Bosu, M. F., and MacDonell, S. G. (2015). On satisfying the android os community: User feedback still central to developers' portfolios. In *Proc. of the 2015 24th Australasian Software Engineering Conference*, pages 78–87. IEEE.

Ma, S., Wang, S., Lo, D., Deng, R. H., and Sun, C. (2015). Active semi-supervised approach for checking app behavior against its description. In *Proc. of the 2015 IEEE 39Th annual computer software and applications conference*, volume 2, pages 179–184. IEEE.

Maalej, W. and Pagano, D. (2011). On the socialness of software. In *Proc. of the 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing*, pages 864–871. IEEE.

Malik, H., Shakshuki, E. M., and Yoo, W.-S. (2020). Comparing mobile apps by identifying 'hot'features. *Future Generation Computer Systems*, 107:659–669.

Nakagawa, H. and Honiden, S. (2023). MAPE-K loop-based goal model generation using generative ai. In *13th International Workshop on Model-Driven Requirements Engineering (MoDRE)*. IEEE.

OpenAI (2023). Chatgpt. `https://chat.openai.com/chat`.

Oriol, M., Stade, M., Fotrousi, F., Nadal, S., Varga, J., Seyff, N., Abello, A., Franch, X., Marco, J., and Schmidt, O. (2018). FAME: supporting continuous requirements elicitation by combining user feedback and monitoring. In *Proc. of the 2018 ieee 26th international requirements engineering conference (re)*, pages 217–227. IEEE.

Pagano, D. and Maalej, W. (2013). User feedback in the appstore: An empirical study. In *Proc. of the 2013 21st IEEE international requirements engineering conference (RE)*, pages 125–134. IEEE.

Papadimitriou, C. H., Raghavan, P., Tamaki, H., and Vempala, S. (2000). Latent semantic indexing: A probabilistic analysis. *Journal of Computer and System Sciences*, 61(2):217–235.

Ren, S., Nakagawa, H., and Tsuchiya, T. (2022). Goal model structuring based on semantic correlation of user reviews. *Intelligent Decision Technologies*, 16(4):737–748.

Seyff, N., Graf, F., and Maiden, N. (2010). Using mobile re tools to give end-users their own voice. In *Proc. of the 2010 18th IEEE International Requirements Engineering Conference*, pages 37–46. IEEE.

Szmrecsanyi, B. (2012). *Grammatical variation in British English dialects: A study in corpus-based dialectometry*. Cambridge University Press.

Zhong, X., ShafieiBavani, E., and Jimeno Yepes, A. (2020). Image-based table recognition: data, model, and evaluation. In *Proc. of the European Conference on Computer Vision*, pages 564–580. Springer.

Zowghi, D. (2018). "Affects" of user involvement in software development. In *Proc. of the 2018 1st International Workshop on Affective Computing for Requirements Engineering (AffectRE)*, pages 13–13. IEEE.