UNIVERSITY OF TARTU
Institute of Computer Science
1632

How Effectively Do LLMs Extract Feature-Sentiment Pairs from App Reviews?

Faiz Ali Shah, Ahmed Sabir, Rajesh Sharma,
Dietmar Pfahl

Software Engineering Analytics Group
Institute of Computer Science
University of Tartu, Tartu, Estonia.

Introduction

- App users provide feedback on the app's functionality by submitting reviews through app marketplaces.
- Analyzing this feedback can help app developers understand users' perceptions of app features and their evolving needs.
- Generating automatic summaries of user sentiments at the level of app features is a technique adopted by researchers.

Introduction

After a new release of an application, feature-sentiment summaries of user reviews can help developers prioritize maintenance efforts.

Sample review

The photo editing tools are fantastic, especially the filters, but the app crashes occasionally when exporting high-resolution images.

feature-sentiment summary

App feature	Sentiment
photo editing tools	Positive
filters	Positive
exporting high-resolution images	negative

Figure 1. Feature-sentiment summary

Background

- Rule-based and supervised methods [3] are used to extract app features, and then relied on sentiment prediction tools.
- Fine-tuning of pre-trained models has significantly outperformed rule-based approaches.
- Recently, LLMs such as ChatGPT has shown the ability to generalize to new tasks without requiring task-specific fine-tuning [4].
- LLMs (with RHLF) have been proven effective in following instruction on tasks with zero-shot or few-shot learning.

Background

Zero-Shot prompt: LLM's ability to perform a task without any prior knowledge related to that specific task.

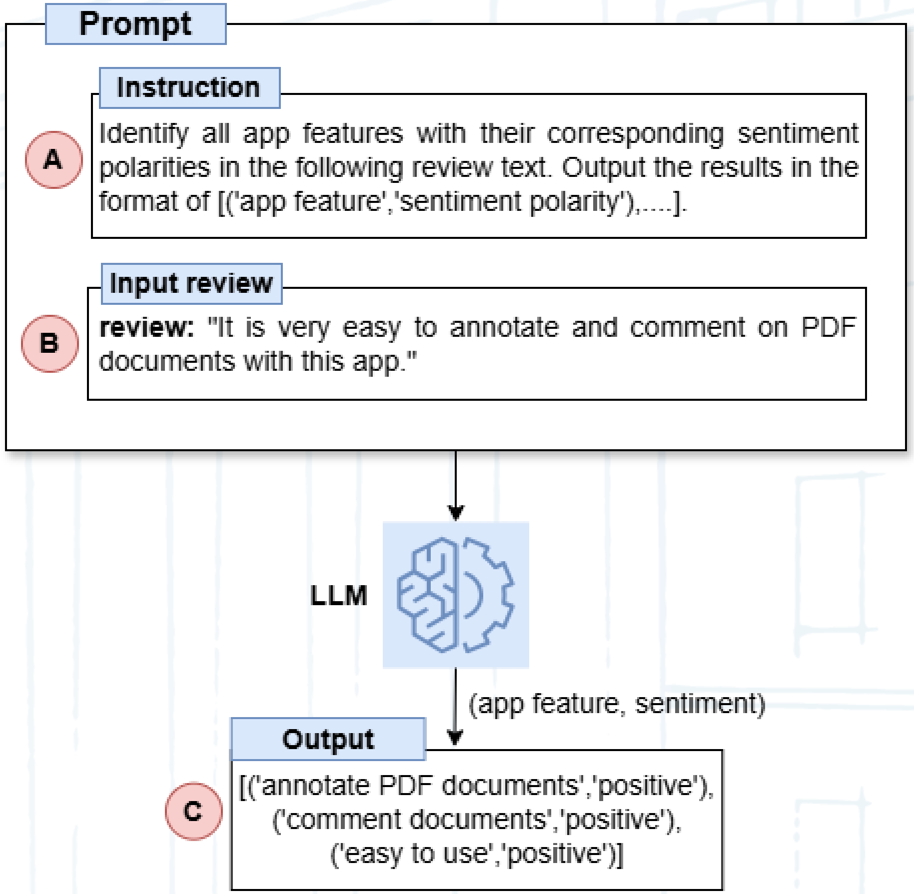


Figure 2. Zero-shot prompt

Background

Few-shot prompt: LLMs are instructed to carry out a task by demonstrating a few examples and a single unlabeled example.

UNIVERSITY OF TARTU
Institute of Computer Science

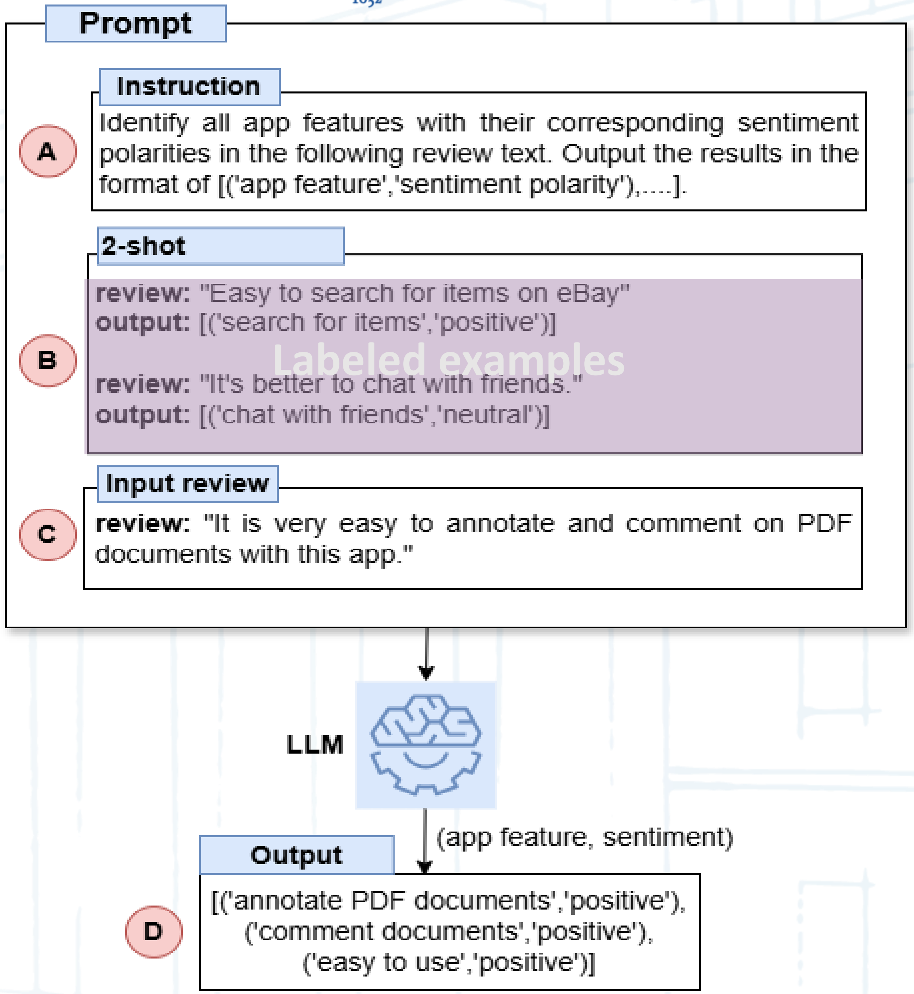


Figure 3. Few-shot prompt

Research Questions (RQs)

UNIVERSITY OF TARTU
Institute of Computer Science

RQ1: How does the *zero-shot* performance of LLMs compare to existing methods for extracting feature-sentiment pairs from app reviews?

RQ2: How does the *few-shot* performance of LLMs compare to *zero-shot* and existing methods for extracting feature-sentiment pairs from app reviews?

Experimental Setup

Labeled dataset [6]
- 1000 user reviews - Eight different applications - 1,521 manually labeled feature-sentiment pairs
Baseline methods
- GuMA [2] - SAFE [1] - ReUS [5] - RE-BERT [3]
Experimented LLMs
- ChatGPT 3.5 Turbo - GPT-4 - Llama-2 Chat (7B, 13B,70B)
Prompting strategy
- Zero shot (RQ1) - Few-shot (1-shot ,5-shot) (RQ2)
Evaluation method
- Token-based exact matching - Toke-based partial matching (by difference of two words)
Performance metrics
- precision, recall, and f1-score

Short prompt (S-Prompt)

As an expert information extractor, identify all app features with their corresponding sentiment polarities (i.e., positive, negative or neutral) in the following review text (enclosed in double quotations). Output the results in the format of [(‘app feature’,’sentiment polarity’),…]. If no app feature is identified, return an empty Python list. Don’t output any other information.

Long prompt (L-Prompt)

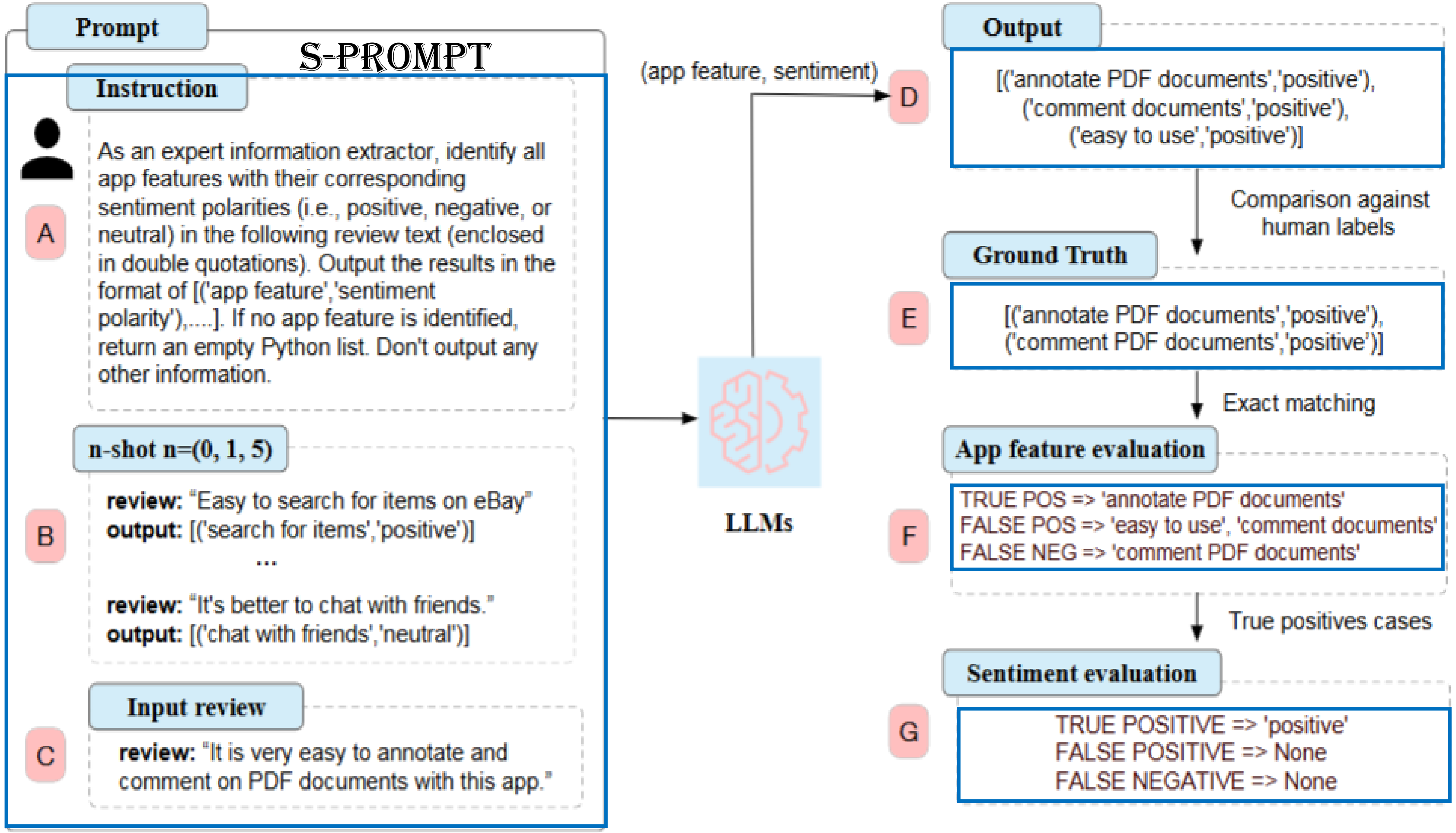
Consider the following definitions of “feature”, “feature expression” and “sentiment polarity”:

- The “feature” refers to a software application functionality (e.g., “send message”), a module(e.g., “user account”) providing functionalities (e.g., “delete account” or “edit information”) or a design component (e.g., UI) providing functional capabilities.
- The “feature expression” is an actual sequence of words that appears in a review text and explicitly indicate a feature.
- The “sentiment polarity” refers to the degree of positivity, negative or neutrality expressed towards the feature of a software application, and the available polarities include: “positive”,” neutral”, “negative”.

As an expert information extractor, identify all feature expressions with their corresponding sentiment polarities (i.e., positive, negative or neutral) in the following review text (enclosed in double quotations). Output the results in the format of [(‘app feature’,’sentiment polarity’),…]. If no app feature is identified, return an empty Python list. Don’t output any other information.

Evaluation Approach

UNIVERSITY OF TARTU
Institute of Computer Science



Three iterations of each LLM have been performed on the entire labeled dataset.

Results (RQ1)

Table 1: Comparison of the *zero-shot* performance of LLMs and baseline methods for extracting app features

Model	Prompt type	Exact match (n = 0) Prec	Exact match (n = 0) Rec	Exact match (n = 0) F1	Partial match 2 ($n = 2$) Prec	Partial match 2 ($n = 2$) Rec	Partial match 2 ($n = 2$) F1
GuMa [11]	-	0.05	0.13	0.07	0.18	0.44	0.25
SAFE [12]	-	0.06	0.06	0.06	0.33	0.34	0.33
ReUS [9]	-	0.08	0.08	0.08	0.33	0.25	0.25
RE-BERT* [8]	-	-	-	0.46	-	-	0.62
ChatGPT [18]	S	0.227 ± 0.03	0.406 ± 0.04	0.290 ± 0.04	0.346 ± 0.04	0.620 ± 0.04	0.443 ± 0.04
ChatGPT	L	0.219 ± 0.04	0.433 ± 0.05	0.290 ± 0.04	0.326 ± 0.04	0.648 ± 0.04	0.433 ± 0.04
GPT-4 [19]	S	**0.257 ± 0.04**	0.404 ± 0.06	0.313 ± 0.05	**0.410 ± 0.05**	0.644 ± 0.06	**0.500 ± 0.05**
GPT-4	L	0.240 ± 0.04	**0.466 ± 0.05**	**0.316 ± 0.05**	0.373 ± 0.06	**0.723 ± 0.06**	0.491 ± 0.06
Llama-2-7B Chat [15]	S	0.157 ± 0.03	0.295 ± 0.05	0.205 ± 0.03	0.255 ± 0.03	0.479 ± 0.05	0.332 ± 0.03
Llama-2-7B Chat	L	0.124 ± 0.01	0.298 ± 0.04	0.175 ± 0.02	0.202 ± 0.02	0.485 ± 0.04	0.285 ± 0.02
Llama-2-13B Chat	S	0.177 ± 0.04	0.265 ± 0.06	0.212 ± 0.05	0.280 ± 0.05	0.420 ± 0.07	0.336 ± 0.06
Llama-2-13B Chat	L	0.141 ± 0.02	0.276 ± 0.04	0.187 ± 0.03	0.231 ± 0.03	0.452 ± 0.06	0.305 ± 0.03
Llama-2-70B Chat	S	0.218 ± 0.05	0.192 ± 0.03	0.202 ± 0.03	0.337 ± 0.05	0.300 ± 0.04	0.314 ± 0.04
Llama-2-70B Chat	L	0.248 ± 0.06	0.273 ± 0.04	0.259 ± 0.05	0.381 ± 0.05	0.422 ± 0.05	0.399 ± 0.04

GPT-4 surpasses SAFE by 17% in f1-score. However, the fine-tuned RE-BERT outperforms GPT-4 by 12% in f1-score.

Results (RQ1)

- To predict positive and neutral sentiments, GPT-4 achieves the best f1-scores of 76.1% and 38.1%, respectively.
- While Llama-2-70B yield the best f1-score of 50.4% for negative sentiment prediction.

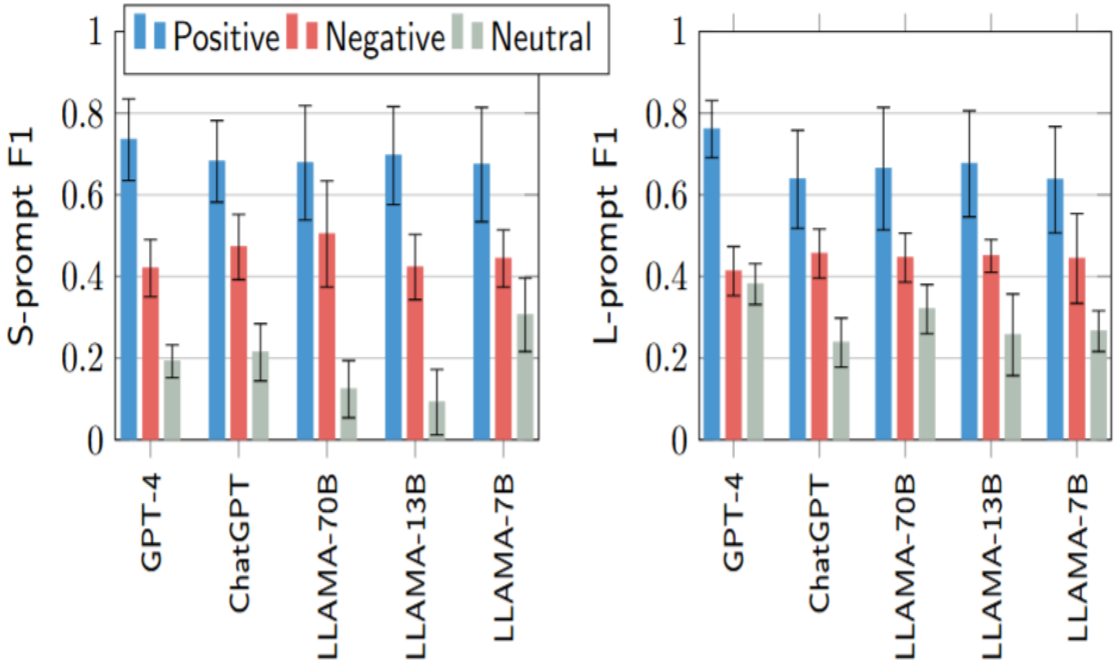


Figure 5: *Zero-shot* performance of LLMs in predicting feature-specific sentiment

Results (RQ2)

UNIVERSITY OF TARTU
Institute of Computer Science

Table 2: Comparison of *few-shot* (i.e. 1-shot and 5-shot) LLM performance against *zero-shot* and baseline methods for extracting app features from user reviews.

Model	Shot	Exact match ($n = 0$)			Partial match 2 ($n = 2$)		
		Prec	Rec	F1	Prec	Rec	F1
GuMa [11]	-	0.05	0.13	0.07	0.18	0.44	0.25
SAFE [12]	-	0.06	0.06	0.06	0.33	0.34	0.33
ReUS [9]	-	0.08	0.08	0.08	0.33	0.25	0.25
RE-BERT* [8]	-	-	-	0.46	-	-	0.62
ChatGPT [18]	0	0.227 ± 0.03	0.406 ± 0.04	0.290 ± 0.04	0.346 ± 0.04	0.620 ± 0.04	0.443 ± 0.04
	1	0.195 ± 0.02	0.402 ± 0.03	0.262 ± 0.03	0.323 ± 0.03	0.668 ± 0.03	0.434 ± 0.04
	5	0.210 ± 0.03	0.370 ± 0.04	0.268 ± 0.03	0.375 ± 0.03	0.662 ± 0.03	0.478 ± 0.03
GPT-4 [19]	0	0.257 ± 0.04	0.404 ± 0.06	0.313 ± 0.05	0.410 ± 0.05	0.644 ± 0.06	0.500 ± 0.05
	1	0.272 ± 0.05	0.437 ± 0.07	0.335 ± 0.06	0.417 ± 0.06	**0.671 ± 0.06**	0.514 ± 0.06
	5	**0.327 ± 0.06**	**0.460 ± 0.06**	**0.382 ± 0.06**	**0.480 ± 0.06**	0.670 ± 0.05	**0.561 ± 0.06**
Llama-2-7B Chat [15]	0	0.157 ± 0.03	0.295 ± 0.05	0.205 ± 0.03	0.255 ± 0.03	0.479 ± 0.05	0.332 ± 0.03
	1	0.172 ± 0.03	0.317 ± 0.05	0.223 ± 0.04	0.269 ± 0.03	0.497 ± 0.05	0.349 ± 0.03
	5	0.197 ± 0.03	0.334 ± 0.05	0.247 ± 0.04	0.312 ± 0.03	0.530 ± 0.05	0.392 ± 0.03
Llama-2-13B Chat	0	0.177 ± 0.04	0.265 ± 0.06	0.212 ± 0.05	0.280 ± 0.05	0.420 ± 0.07	0.336 ± 0.06
	1	0.158 ± 0.02	0.310 ± 0.04	0.209 ± 0.03	0.267 ± 0.02	0.525 ± 0.03	0.354 ± 0.02
	5	0.186 ± 0.02	0.300 ± 0.03	0.229 ± 0.02	0.317 ± 0.02	0.511 ± 0.03	0.391 ± 0.02
Llama-2-70B Chat	0	0.218 ± 0.05	0.192 ± 0.03	0.202 ± 0.03	0.337 ± 0.05	0.300 ± 0.04	0.314 ± 0.04
	1	0.171 ± 0.04	0.329 ± 0.07	0.225 ± 0.05	0.278 ± 0.03	0.535 ± 0.05	0.366 ± 0.04
	5	0.200 ± 0.03	0.383 ± 0.05	0.263 ± 0.04	0.320 ± 0.03	0.614 ± 0.05	0.420 ± 0.03

- With 5-shot learning, GPT-5 improved the f1-score by 6% (i.e., 56%) representing a 23% improvement over the SAFE approach.
- The fine-tuned BERT still outperforms GPT-4 by 6% in f1-score.

Results (RQ2)

UNIVERSITY OF TARTU
Institute of Computer Science

- In positive sentiment prediction, 5-shot improves the f1-score by 7% for GPT-4 and 3% for Llama-70B.
- For negative sentiment, 5-shot does not improve the performance of GPT-4, and f1-score decreases for ChatGPT and Llama-2-70B by 7% and 8%, respectively.
- For neutral sentiment prediction, 5-shot improves the f1-score by 23% for GPT4 and 14% for Llama-2-70B.

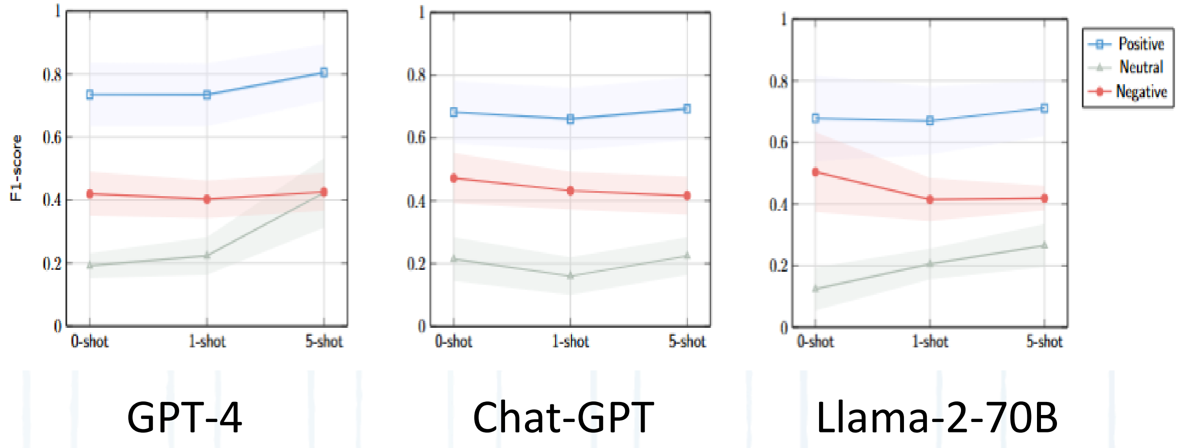


Figure 6: Comparison of *zero-shot, 1-shot, and 5-shot* performances of GPT-4, ChatGPT, and Llama-70B in predicting feature-specific sentiment.

Error Analysis of LLMs

Table 3: Error analysis of feature-sentiment pairs extracted by Llama-2-70B, ChatGPT and GPT-4 from user reviews (R1 to R6).

Review	LLama-2-70B Chat		ChatGPT		GPT-4	
	App feature	Sentiment	App feature	Sentiment	App feature	Sentiment
R1 -> So many bugs. force crashes and [messages cannot be sent]NEU	bugs ✗ force crashes ✗ messages cannot be sent ✓	NEG ✗ NEG ✗ NEG ✓	bugs ✗ force crashes ✗ messages cannot be sent ✓	NEG ✗ NEG ✗ NEG ✓	bugs ✗ force crashes ✗ messages cannot be sent ✓	NEG ✗ NEG ✗ NEG ✓
R2 -> Best app in world	app ✗	POS ✗	best app ✗	POS ✗	None ✓	None ✓
R3 -> Its easy to use and has a good [user interface]POS	easy to use ✗ good user interface ✓	POS ✗ POS ✓	easy to use ✗ good user interface ✓	POS ✗ POS ✓	easy to use ✗ user interface ✓	POS ✗ POS ✓
R4 -> I cant [add filtets w/ pictures]NEU with the latest version using my galaxy.	add filters with pictures ✗	NEG ✗	add filters ✗	NEG ✗	add filters ✗	NEG ✗
R5 -> Need to be [login]NEU & [log out]NEU feature for security reason or [password]NEU option.	login & log out feature ✗ password option ✓	NEU ✓ NEU ✓	login & log out feature ✗ password option ✓	POS ✗ POS ✗	login & log out feature ✗ password option ✓	POS ✗ POS ✗
R6 -> I think they should add is a tempo/speed thing so you can [listen at different speeds]NEU that would be really cool.	tempo/speed control ✗	POS ✗	add tempo/speed control ✗ listen at different speeds ✓	POS ✗ POS ✗	tempo/speed thing ✗	POS ✗

LLMs often confuse neutral sentiment with either negative or positive sentiment.

Threats to validity

UNIVERSITY OF TARTU
Institute of Computer Science

- Prompting LLMs may struggle to understand the full context and intricacies of a given prompt.
- The landscape of LLMs is rapidly evolving, it is yet to be investigated whether the results generalize to other LLMs.
- All models were evaluated with parameters temperate set to zero and maximum output tokens set to 1000.
- Our evaluation study relies on a relatively small labeled dataset comprising 1000 labeled reviews from eight distinct apps.

Conclusions

UNIVERSITY OF TARTU
Institute of Computer Science

- Our evaluation study inform that the precision and recall of LLMs in extracting feature-sentiment pairs are yet not adequate for practical applications.
- Although prompt engineering demonstrates a lower performance than fine-tuned language models, it offers a cost effective approach where the labeled data is scarce.

Future work

UNIVERSITY OF TARTU
Institute of Computer Science

- In few-shot experiments, selecting examples that are semantically similar to the input review may further enhance the performance of LLM models.
- A promising direction is to explore enhancing the effectiveness of fine-tuning by leveraging synthetic datasets.

References

1. Timo Johann, Christoph Stanik, Alireza M.B. Alizadeh, and Walid Maalej. SAFE:A Simple Approach for Feature Extraction from App Descriptions and App Re-views. RE, 2017
2. Emitza Guzman and Walid Maalej. How do users like this feature? A fine grained sentiment analysis of App reviews. RE, 2014.
3. Adailton Ferreira De Araújo and Ricardo Marcondes Marcacini. RE-BERT: Automatic extraction of software requirements from app reviews using BERT language model. Proceedings of the ACM Symposium on Applied Computing, 2021.
4. Yiheng Liu, Tianle Han, Siyuan Ma, Jiayue Zhang, Yuanyuan Yang, Jiaming Tian,Hao He, Antong Li, Mengshen He, Zhengliang Liu, et al. Summary of chatgpt-related research and perspective towards the future of large language models. Meta-Radiology, 2023.
5. Mauro Dragoni, Marco Federici, and Andi Rexha. An unsupervised aspect extraction strategy for monitoring real-time reviews stream. Information processing &management, 2019.
6. Jacek Dabrowski, Emmanuel Letier, Anna Perini, and Angelo Susi. Mining and searching app reviews for requirements engineering: Evaluation and replication studies. Information Systems, 114:102181, 3 2023.

Thank you!

Questions?