2023 IEEE Conference on Artificial Intelligence (CAI) | 979-8-3503-3984-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/CAI54212.2023.00147

A New Method Using LLMs for Keypoints Generation in Qualitative Data Analysis

Fengxiang Zhao[1] Fan Yu[2] Timothy Trull[3] Yi Shang[1]
[1]*Department of EECS, University of Missouri-Columbia, USA*
[2]*School of Information Science and Learning Technogies, University of Missouri-Columbia, USA*
[3]*Department of Psychological Science, University of Missouri-Columbia, USA*
{fzfmx, fybxd, trullt, shangy}@missouri.edu

***Abstract*—Qualitative data analysis (QDA) is useful for identifying patterns, themes, and relationships among data. In this paper, we propose a new method that uses large language models (LLMs), such as GPT-based Models, to improve QDA, in Ecological Momentary Assessment (EMA) studies as an example, by automating keypoints extraction and relevance evaluation. Experimental results on the *IBM-ArgKP-2021* dataset show improved performance of the new method over existing work, achieving higher accuracy while reducing time and effort in the coding process of QDA, and demonstrate the effectiveness of our proposed method in various application settings.**

***Index Terms*—Qualitative data analysis (QDA), ecological momentary assessment (EMA), large language models (LLMs), ChatGPT, keypoints generation**

I. Introduction

Qualitative data analysis (QDA), the process of examining and interpreting non-numerical data, such as text, images, audio, and video, is important for identifying patterns, themes, and relationships among data [1]. Psychology studies, such as Ecological Momentary Assessment (EMA), often use QDA to understand individuals' lived experiences. EMA is a research method used in psychology to gather survey and environmental data from participants about their real-life experiences using mobile devices such as smartphones or wearable sensors. QDA typically has several steps, including data familiarization, coding, theme development, and interpretation.

Coding is a key step of QDA that involves assigning labels or codes to segments of the data that capture specific concepts, ideas, or themes. These codes, which can be descriptive or analytical, are used to organize the data and identify patterns. Keypoints are the essential elements or concepts that must be identified and coded in QDA. Researchers traditionally assign keypoints to data units during a time-consuming coding process [2]. Recently, Large Language Models (LLMs), such as *ChatGPT* (https://chat.openai.com/chat), GPT-3.5 [3] and GPT-4, exhibited remarkable language understanding and generative abilities. In this paper, we propose a new method utilizing LLMs to improve QDA by automating keypoints extraction and relevance evaluation tasks. Our method offers efficient and accurate solutions while reducing time and effort involved in the coding processes. The effectiveness and performance of the method are demonstrated in experiments, highlighting its potential in various areas, including EMA studies, open-ended question analysis, and multi-document summarization.

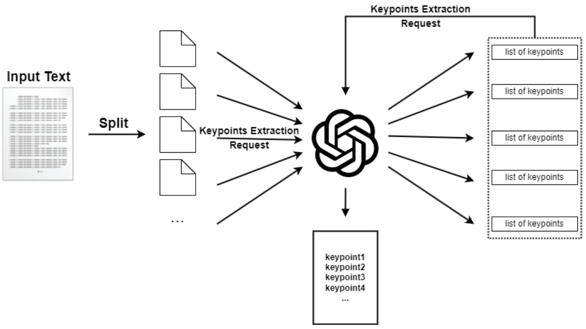

Fig. 1. Keypoints extraction pipeline in the Key Points Extractor.

II. Method

The proposed method comprises two primary components: *Keypoints Extractor* and *Keypoints Relevance Evaluator*. *Keypoints Extractor* identifies keypoints from the Input Text, while *Keypoints Relevance Evaluator* evaluates the degree of relevance of Input Text with respect to the identified keypoints.

Keypoints Extractor, functioning as a multi-document summarization tool, uses ChatGPT to generate keypoints from Input Text, such as user responses to surveys in EMA studies, in two steps. First, ChatGPT generates a set of candidate keypoints. Due to GPT's token limit (4096 for *GPT-3.5* and 8192 for *GPT-4*), long inputs are partitioned into segments within the limit for ChatGPT to generate candidate keypoints from each. Then, ChatGPT refines and extracts the final set of keypoints from the candidates. Fig. 1 shows the pipeline.

Keypoints Relevance Evaluator accepts a text and a list of keypoints as inputs. Let T denote Input Text, K denote the set of keypoints, and $k \in K$ denote a single keypoint. The evaluator computes the degree of relevance for each (T, k) pair. The degree of relevance is expressed as an integer ranging from 1 (not relevant) to 5 (highly relevant), as shown in Fig. 2.

We transform unstructured ChatGPT output into structured data in tabular form containing response-relevance scores of keypoints. We utilize prompt engineering techniques to guide ChatGPT towards generating outputs in the target format.

979-8-3503-3984-0/23/$31.00 ©2023 IEEE
DOI 10.1109/CAI54212.2023.00147

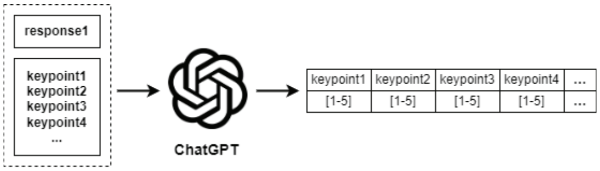

Fig. 2. Response coding pipeline in the Keypoints Relevance Evaluator.

Our implementation features a customizable, user-friendly interface built using *Gradio* [4], where users can upload materials and generate corresponding keypoints. The interface allows for customization of the number of generated keywords and enables users to edit, add, or delete keywords before initiating *Keypoints Relevance Evaluator*. The token usage panel displays estimated ChatGPT API costs, allowing users to manage costs easily.

III. Experiments

We tested our method using the publicly available *IBM-ArgKP-2021* dataset [5], which includes 27,000 pairs of *(argument, key point)* related to 31 debatable topics. Each pair, classified as matching or non-matching, includes the argument's pro/con stance. We conducted our experiments using both *GPT-3.5* and *GPT-4* models.

A. Experiments Setup

Two experiments were conducted to examine the performance of our method.

1) Keypoints Extraction: We utilized *key_points* provided in the dataset as reference and used the *Keypoints Extractor* to generate a list of keypoints for each topic. We also used randomly sampled arguments (Random) and MEAD [6], an extractive summarizer, as the baselines for comparison. To assess the quality of extracted keypoints, we utilized four ROUGE scores: *ROUGE-1*, *ROUGE-2*, *ROUGE-L*, and *ROUGE-W* [7]. These scores measured the overlap between generated and reference keypoints based on unigrams, bigrams, and the longest common subsequence. We set the weight in *ROUGE-W* to 1.2 in our evaluation.

2) Mapping Arguments to Keypoints: We randomly selected 100 arguments from each topic, resulting in a total of 3,100 arguments. We then utilized the Keypoints Relevance Evaluator to identify the mapping of each argument to all possible *key_points* from the reference dataset within the same topic. For each generated matching pair *(argument, keypoint)*, we compared it with the label in the dataset to determine if it was a correct match.

B. Results

1) Keypoints Extraction: Table I compares the various keypoints extraction methods. ROUGE scores are averaged over all topics. *GPT-3.5* outperforms the other models in all cases. *GPT-4* has slightly lower scores. Both GPT-based models show promising results and are more effective at generating summaries that are similar to reference summaries compared to the MEAD and random methods.

2) Mapping Arguments to Keypoints: Table II compares performances of five different methods for mapping arguments to keypoints. In addition to our proposed method using *GPT-3.5* and *GPT-4*, we include results from IBM Research's best supervised model *BERT-large*, and best unsupervised model, *BERT Embeddings* [5]. The performance metrics include accuracy, precision, recall, and F1 scores. *GPT-4* outperforms all other models, surpassing BERT-based state-of-the-art performance. This demonstrates *GPT-4*'s ability to handle complex tasks using prompts alone and its potential for zero-shot learning.

TABLE I
Performance comparison of four different keypoints extraction methods based on average ROUGE scores.

Method	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-W
MEAD	0.059	0.015	0.051	0.102
Random	0.118	0.028	0.108	0.152
GPT-3.5	**0.134**	**0.039**	**0.124**	**0.169**
GPT-4	0.128	0.034	0.116	0.167

TABLE II
Performance comparison of five different methods for mapping arguments to keypoints.

	Accuracy	Precision	Recall	F1
Random Predictions	0.495	0.202	0.502	0.288
BERT Embeddings	0.660	0.319	0.550	0.403
BERT-large (ArgKP)	0.868	0.685	0.688	0.684
GPT-3.5	0.722	0.371	0.531	0.436
GPT-4	**0.944**	**0.868**	**0.856**	**0.862**

In summary, our preliminary experimental results demonstrate our proposed method's effectiveness, highlighting the potential of LLMs for various application settings, while reducing time and effort in the coding process of QDA.

References

[1] M. Savin-Baden *et al.*, "Qualitative research: The essential guide to theory and practice," 2012.

[2] H. H. AlYahmady and S. S. Alabri, "Using nvivo for data analysis in qualitative research," *International Interdisciplinary Journal of Education*, vol. 2, pp. 181–186, 2013.

[3] T. Brown and et al, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901.

[4] A. Abid *et al.*, "Gradio: Hassle-free sharing and testing of ml models in the wild," *arXiv preprint arXiv:1906.02569*, 2019.

[5] R. Bar-Haim *et al.*, "From arguments to key points: Towards automatic argument summarization," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 4029–4039. [Online]. Available: https://aclanthology.org/2020.acl-main.371

[6] D. R. Radev, H. Jing, and M. Budzikowska, "Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies," 2000.

[7] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Annual Meeting of the Association for Computational Linguistics*, 2004.