Interactive Question Answering for Multimodal Lifelog Retrieval

Ly-Duyen Tran[1](✉), Liting Zhou[1], Binh Nguyen[2,3], and Cathal Gurrin[1]

[1] Dublin City University, Dublin, Ireland
allie.tran@adaptcentre.ie
[2] AISIA Research Lab, Ho Chi Minh, Vietnam
[3] Ho Chi Minh University of Science, Vietnam National University, Hanoi, Vietnam

Abstract. Supporting Question Answering (QA) tasks is the next step for lifelog retrieval systems, similar to the progression of the parent field of information retrieval. In this paper, we propose a new pipeline to tackle the QA task in the context of lifelogging, which is based on the open-domain QA pipeline. We incorporate this pipeline into a multimodal lifelog retrieval system, which allows users to submit questions prevalent to a lifelog and then suggests possible text answers based on multimodal data. A test collection is developed to facilitate the user study, the aim of which is to evaluate the effectiveness of the proposed system compared to a conventional lifelog retrieval system. The results show that the proposed system is more effective than the conventional system, in terms of both effectiveness and user satisfaction. The results also suggest that the proposed system is more valuable for novice users, while both systems are equally effective for experienced users.

Keywords: Lifelogging · Question answering · Human-Computer Interaction

1 Introduction

Lifelogging has gained significant attention in recent years as a means of digitally recording and preserving one's life experiences. Lifelog data typically consists of multimodal information, including text, images, audio, and video, and presents unique challenges for efficient retrieval and organisation due to the volumes of multimodal data collected. To address these issues, pioneering research challenges and competitions have been organised, such as Lifelog Search Challenges [26] and NTCIR Lifelog tasks [34]. The Lifelog Search Challenge (LSC) is one of the most well-known benchmarking activities in the field, which has been running annually since 2018, with the aim of advancing the state-of-the-art in lifelog retrieval systems in an open, metrics-based manner. The challenge focuses on comparing novel techniques for supporting users to efficiently search for a specific moment in their lifelogs. However, most research in the area has heretofore focused on interactive retrieval systems, while the Question Answering

S. Rudinac et al. (Eds.): MMM 2024, LNCS 14565, pp. 68–81, 2024.
https://doi.org/10.1007/978-3-031-56435-2_6

(QA) challenge remains an under-explored topic. With the increasing prevalence of pervasive computing, there is a need to integrate question answering (QA) capabilities into lifelog retrieval systems, allowing users to ask specific questions about their lifelogs and receive text-based or spoken answers. Understanding this need, a QA task was introduced at LSC'22 [9], accepting images as the answers, and thus it was not a true QA task. In LSC'23 [10], the QA task was fully integrated, meaning that QA topics were answered by text-based submissions. This highlights the direction of the research community, which is moving towards supporting lifelog QA systems.

Although a lifelog QA dataset, LLQA [24], was constructed in order to gain more attention for the task, all questions in this dataset are pertained to short, provided snippets of lifelog data. In other words, the questions are related to a particular activity at a point in time, for example, 'What is the lifelogger holding in his hand [at this particular time]?'. In reality, more open-ended type of questions that span any period of time, from a moment to a lifetime, are more prevalent, such as 'What is my favourite drink?' or 'Where did I go on holiday last summer?'. Therefore, a comprehensive question dataset is therefore necessary to address this issue. In this paper, we present a test collection for the QA task, utilising all the published datasets in the LSC spanning from 2016 to 2023. By leveraging the existing LSC datasets, we aim to create a comprehensive test collection that addresses the broader range of lifelog queries users may have. Moreover, we propose a pipeline to incorporate QA capabilities into an existing lifelog retrieval system. We conduct a user study to evaluate the effectiveness and user satisfaction of our proposed lifelog QA system and compare it to a baseline search-only approach. Our preliminary results demonstrate the superiority of our proposed system over the baseline approach, with significant improvements in both effectiveness and user satisfaction metrics. Furthermore, the results suggest that our proposed system is particularly well-suited for novice users, offering a more intuitive and efficient lifelog retrieval experience. The following sections of this paper will describe the details of our proposed pipeline, the construction of the test collection, the user study methodology, and the comprehensive analysis of the results obtained.

The contributions of this paper are thus as follows: (1) a novel pipeline for multimodal lifelog QA systems, drawing inspiration from the open-domain QA pipeline; (2) a test collection for the lifelog QA task comprising 235 questions sourced from the LSC datasets; and (3) a user study assessing the effectiveness and user satisfaction of our lifelog QA system in comparison to a search-only baseline approach.

2 Related Work

2.1 Lifelog Retrieval

Retrieval systems for lifelog data have been a popular research topic for almost a decade now, since the first NTCIR-lifelog challenge in 2015 [8]. It is a crucial task in order to manage and make use of the large amount of multimodal data collected by lifeloggers. The seminal lifelog retrieval system is MyLifeBits [7], which supported limited full-text search, text and audio annotations, and hyperlinks. Since then, many other lifelog retrieval systems have been proposed and evaluated. The Lifelog Search Challenge (LSC) is an annual benchmarking campaign that aims to advance the state-of-the-art in lifelog retrieval systems. The dominant approach of the participating teams has been focusing on concept-based techniques, leveraging computer vision models to automatically extract visual analysis from lifelog images, such as object recognition, scene understanding, and optical character recognition (OCR). The outputs of these models, also known as 'concepts', are then used in accompanying metadata (e.g. timestamps, GPS coordinates, etc.) for indexing and retrieval. Various ranking techniques borrowed from the field of text-based information retrieval have been explored, such as TF-IDF [28], BM25 [3,29], bag-of-words (BoW) [19] to rank the lifelog moments based on the concepts. Other metadata such as timestamps and location information are also used to improve the retrieval performance by boolean filtering [23] or map visualisation [28]. Recently, with the rise of cross-modal embedding models, such as CLIP [20] and CoCa [33], large-scale pretrained models have been utilised to extract the visual and textual features from image contents and questions, and then provide a similarity score between the features to rank the lifelog moments. This embedding-based approach allows a more user-friendly experience by allowing users to search for lifelog data using natural language queries and significantly improves the retrieval performance [1,27]. As a result, most conventional search tasks in the LSC are considered mostly solved by this embedding-based approach. This allowed the organisers to introduce the lifelog QA task in the LSC'22, aiming to evaluate the effectiveness of lifelog retrieval systems in answering questions about lifelog data. Since QA is a relatively new task in the lifelogging domain, there is a lack of research in this area. Our study aims to contribute to this area by proposing a pipeline for integrating QA capabilities into lifelog systems and evaluating its effectiveness compared to a baseline search-only approach.

2.2 Open-Domain Question Answering

Open-domain QA (OpenQA) is the task of answering questions without any specified context, as opposed to machine reading comprehension (MRC) where specified context passages are provided. Most modern OpenQA systems follow a *'Retriever-Reader'* architecture [4,35] which contains a *Retriever* and a *Reader*. Given a question, the *Retriever* is responsible for retrieving relevant documents to the question in an open-domain dataset such as Wikipedia and the World Wide Web (WWW); while the *Reader* aims at inferring the final answer from the received documents, which is usually a neural MRC model. Specifically, the *Retriever* can utilise traditional information retrieval techniques such as TF-IDF [4] and BM25 [32], or more advanced deep retrieval models to encode the question and documents [12,15]. After that, approaches for the *Reader* can be categorised into extractive and generative models. Extractive models [4,12,32] are designed to extract an answer span from the retrieved documents using

BERT [5], RoBERTA [18], etc. On the other hand, generative approaches [11,17] apply models such as BART [16] and T5 [21] to generate the answer in an open-ended manner. To further extend the architecture, some works [14,30] have proposed to re-rank the retrieved documents before feeding them into the *Reader* [14], or train the entire OpenQA system in an end-to-end manner [15,17].

In this paper, we take inspiration from open-domain QA research to design a lifelog QA system for the following reasons: (1) the lifelog QA task is similar to the open-domain QA task in the way that the questions are not limited to a specific event or image, but the whole lifelog; and (2) the two-stage architecture is flexible enough to incorporate with existing state-of-the-art lifelog retrieval systems without the need to re-train the entire system.

3 QA Test Collection

In order to compile a comprehensive QA test collection, we utilise the largest two lifelog datasets in the LSC, namely LSC'21 [26] and LSC'22 [9]. Together, these datasets feature an extensive repository of lifelogging data collected by one lifelogger. This data encompasses various types of multimodal information, including over 900,000 point-of-view images, music listening history, biometrics, and GPS coordinates.

As the time of writing, there are 19 official QA information needs (topics) posed by the lifelogger who created the datasets for the LSC challenge (8 in LSC'22 and 11 in LSC'23). In addition to these, we have created a larger collection of topics to include more variety in the user study, leading to 235 questions in total. These questions were inspired by the official known-item search (KIS) topics in all LSCs from 2019 to 2023. An example KIS topic is '*I was building a computer alone in the early morning on a Friday at a desk with a blue background. Sometimes I needed to refer to the manual. I remember some Chinese posters on the desk background. I was in Dublin City University in 2015*'. For each topic, we identified the relevant lifelog data that were provided by the organisers, including time, location, and lifelog images. We then created questions based on the information in the topic description and the provided data. For example, one question for the above topic is *'How many days did it take for me to build my computer back in March 2015?'*, whose answer, *'2 d'*, can be found by looking at the timestamps of the ground-truth images. After that, each question in the collection is labelled based on the type of information that is asked, such as Location, Time, and Colour. The test collection focuses on questions that have specific answers, which are either a single word or a short phrase, with as little ambiguity as possible. This is to ensure that the answers can be easily evaluated. The questions are also designed to be as diverse as possible, to cover different types of information that can be retrieved from a lifelog. Thus, we propose 8 different types of questions for this collection as follows:

- **Location**: these are questions that ask about the name of a country, a city, or a venue (e.g. restaurant) where some specific events happened. For example, 'Where did I go the get my car repaired in 2020?';

- **Object**: the answers generally refer to some objects that are involved in the events. For example, ‘What did I eat for dinner on the 1st of January 2020?’
- **Counting**: these require counting the number of people or things that appeared in an event. For example, ‘How many different papers did I read on the plane going to Germany back in June?’
- **Time**: these are questions that ask about the date/time of some events. For example, ‘When did I last go to the zoo?’ or ‘What time did I go shopping for emergency supplies in 2020?’
- **Frequency**: these require counting the number of times some activities happened. For example, ‘How many times did I have BBQs in my garden in the summer of 2015?’
- **OCR**: the answers are some texts that appeared in the lifelog images. For example, ‘Which airline did I fly with most often in 2019?’ requires reading the boarding passes or the airlines’ brochures on the back of the seats.
- **Colour**: these are questions that ask about the colour of some objects. For example, ‘What colour was the rental car I drove before 2018?’
- **Duration**: the answers are the duration of some events. For example, ‘How long did it take me to drive from Dublin to Sligo in 2016?’

The distribution of the questions in the collection is shown in Fig. 1. Time and Location are the most common types of questions, which is to be expected. The least common type is Frequency, which possibly is because it is difficult to verify the answer in a short time, which is not suitable for the user study. The full list of questions and their answers is available at https://docs.google.com/spreadsheets/d/1eTlKfurPg0LOT-PDkf3SpctdkvrlyV_u1v3IOdgU4wU.

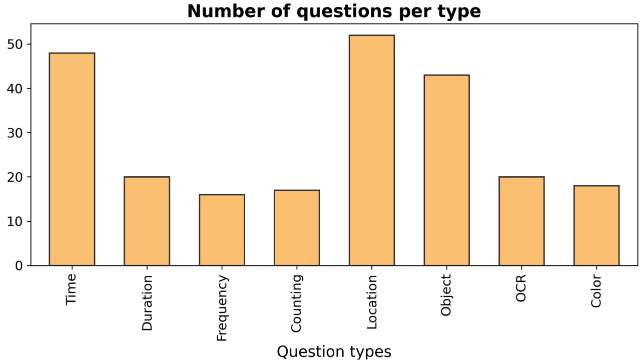

Fig. 1. Distribution of the questions in the test collection.

4 Lifelog Question Answering Pipeline

Inspired by the open-domain QA pipeline [4], we formally propose a pipeline for the lifelog QA system as shown in Fig. 2. Two key components of the pipeline are

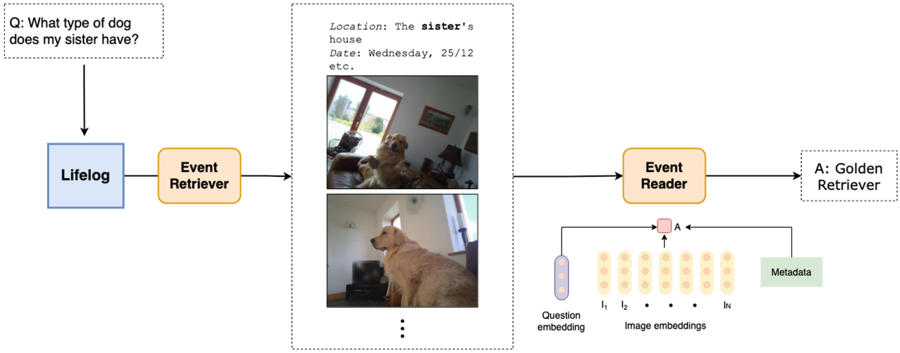

Fig. 2. The proposed pipeline for the QA system. The Event Retriever is in charge of retrieving the lifelog data that are relevant to the given question. On the other hand, the Event Reader component is responsible for generating answers based on the retrieved data.

(1) Event Retriever and (2) Event Reader. The Event Retriever is in charge of retrieving the lifelog data that are relevant to the given question. On the other hand, the Event Reader component is responsible for generating answers based on the retrieved data. This pipeline is designed to be flexible so that different retrieval and QA methods can be used. As a result, it can seamlessly integrate with most existing lifelog retrieval systems, serving as the initial component in the process.

4.1 Event Retriever

The first component is a crucial part of the pipeline, as it is responsible for retrieving the relevant lifelog information that is used to generate the answers. Given a question, the lifelog retrieval component determines the relevance of events in the lifelog to the question based on various multimodal features, such as time, location, and image content. The events are then ranked using a suitable ranking method as seen in a conventional lifelog retrieval system, namely boolean filtering, text-based retrieval, or embedding-based retrieval as described in Sect. 2.1.

To adapt conventional image-focus lifelog retrieval systems, a post-processing step might be useful to aggregate the information from the retrieved data and reduce the amount of information to be passed to the question answering component, which is important for the efficiency of the system. Grouping data that belong to the same event is a possible approach, which can be done by clustering the retrieved events based on their time and location information.

Our proposed system is built upon MyEachtra[25], which participated in LSC'23 and achieved the second-best overall performance. Location and time information are extracted directly from the question and used to filter the events. The remaining part of the question is encoded by the text encoder from OpenAI CLIP [20] and is used to rank the events based on similarity scores. The main difference between MyEachtra from other conventional lifelog retrieval systems is that it expands the unit of retrieval from point-in-time moments to a longer period of time, or 'events', aiming to reduce the search space and provide more lifelog context to the user. This also allows the system to support more complex queries, such as questions about duration and frequency, which are difficult to answer without any organisation of the lifelog data. Since MyEachtra is event-focus, the post-processing step described above is not necessary.

The top-ranked events are then passed to the question answering component to generate the answers. The cut-off point for the number of events to be passed to the question answering component is a hyperparameter of the system, which can be tuned to achieve the best performance. It is also important to note that different types of questions may require different numbers of events to be passed to the question answering component. For example, questions that require counting the Frequency of some events may require more events to be passed to the question answering component than questions that ask about the Location of some events. In this paper, we use the top 10 events as the default cut-off point for all types of questions to simplify the process. However, this can be adjusted in the future to improve the performance of the system.

4.2 Event Reader

This QA component of the pipeline is responsible for generating the answers based on the retrieved events. The answers are generated by combining the information from the retrieved events and the question. The information from the retrieved events can be extracted from the metadata, such as time and location, or the image content, such as OCR text. To address the multimodality of the lifelog data, we propose an ensemble of two different models to handle both visual and non-visual information. The original MyEachtra system proposed using video QA models and treating each event as a video clip with a very low frame rate. This allows the system to leverage both the visual content and the temporal relationship between the images in the events. However, this model is not suitable for questions that do not require visual information, such as questions about Time and Location. To address this issue, we propose to add a text-only QA model to handle non-visual information. Finally, the two models are combined to generate the suggested answers which are shown to the user.

Specifically, in this paper, FrozenBiLM [31] is employed as the VideoQA model, which builds on a frozen bidirectional language model as well as a frozen visual encoder, CLIP [20]. FrozenBiLM was pretrained on a large-scale video-caption pairs dataset WebVid10M [2]. As it builds on a language model, FrozenBiLM can be used to predict the most probable answer given the question as a masked prompt, such as '[CLS] Question: <Question>? Answer: [MASK]'. We also experimented on finetuning FrozenBiLM on the LLQA dataset [24], however, the performance does not improve due to the small size of the dataset. Thus, we use the model that was fine-tuned on the ActivityNet-QA dataset [6] instead.

The new addition to the model is the use of the text-only QA model to handle non-visual information. Related information from the metadata is used to generate a contextual paragraph in the format of 'The event happened at <location> on <date>, starting at <time> and ending at <time>. Text that can be read from the images includes: <OCR text>'. We use the RoBERTA model [18], pretrained on SQuAD 2.0 [22] to predict the answer span from the generated paragraph.

5 User Study

To evaluate the effectiveness of the proposed lifelog QA system, we conducted a user study comparing the performance of the QA system to a baseline search-only system. This allows for a direct comparison between the two systems, providing insights into the effectiveness of the QA system and the potential to improve the lifelog retrieval experience.

5.1 Setup

A total of 10 participants, with ages ranging from 20 to 35, were recruited for the user study. All participants have basic computer skills, with very little familiarity with the concept of lifelog retrieval and question answering. The participants were randomly to one of the two groups: the baseline group and the QA group. The baseline group was asked to use the baseline system first, then the QA system, and vice versa for the QA group. This is to ensure that the order of the systems does not affect the results.

Each participant had a training period of 10-15 min to get familiar with the concept of lifelogging and the systems before the test. For each system, the participants were asked to use the system to answer 8 randomly selected questions from the test collection, one for each type of question. Three minutes were given for each question. If the participants were sure about the answer, they could submit it and the judging system (controlled by a real-time human judge) would inform them whether the answer was correct or not. If the answer was incorrect, the participants were asked to try again. If they could not find the answer within 3 min, they were asked to move on to the next question. The participants were also asked to fill in a questionnaire after using each system. The questionnaire is based on the User Experience Questionnaire (UEQ) [13], which is a standard questionnaire for evaluating the usability of a system. The questionnaire consists of 8 questions, each of which is rated on a scale of -3 to 3 (with 0 as the neutral score). Feedback on the system was also encouraged, which is used to improve the system in the future.

Similarly to the LSC, the performance of the systems is measured based on (1) the accuracy of the answers, (2) the number of wrong submissions, w, and (3) the time taken to answer the questions, t. For each task, if it is solved (the correct answer was submitted), the score s is calculated as follows:

$$s = 100 - 50 \times \frac{t}{180} - 10 \times w \tag{1}$$

If the task is not solved, $s = 0$.

5.2 Baseline System

The baseline system used in this user study is a lifelog retrieval system that is also the LSC'23 baseline system, E-Myscéal [27], an embedding-based variation of the original Myscéal [28]. Myscéal and its upgraded versions have been the best-performing system in the LSC since 2020 and participated in LSC'23 as a baseline system for benchmarking other lifelog systems. It is designed to accommodate novice users by accepting full sentences as search queries. A query parsing component is used to extract the relevant information from the query, such as location, time, and visual information. The extracted information is then used to compose Elasticsearch queries to retrieve the relevant images. The retrieved images are then ranked based on their relevance to the query. The mechanism to retrieve the textual data field is BM25, while the mechanism to retrieve the visual data field is the cosine similarity between the query and the image features. The query and image features are extracted using the OpenAI CLIP model [20].

The user can also browse the lifelog images using a popover timeline, which is shown when the user clicks on any image shown on the result page. The popover timeline shows the images taken before and after the selected image, which allows the user to browse the images in chronological order. The user can also click on any image in the popover timeline to view the image in full size. More features to support the user in the lifelog retrieval task are also provided, such as the ability to search for visually similar images, filter the results by map location, and most importantly, search for temporally related queries.

5.3 QA System

We use the proposed pipeline to integrate QA capabilities into the Myscéal system by (1) shifting the unit of retrieval to events, which is the main difference between MyEachtra and E-Myscéal [28] in the retrieval stage; and (2) adding a QA component to generate the answers based on the retrieved events. Refer to Sect. 4 for more details about the pipeline.

6 Results

6.1 Overall Score

The overall score of each system is calculated as the average score of all the tasks. The results are shown in Fig. 3. The QA system outperforms the baseline system in terms of the overall score. The average score of the QA system is 69.78, while that of the baseline system is 64.96. However, the average wrong submissions and time taken by both systems are not significantly different. The average wrong submissions of the QA system is 0.42, while that of the baseline system is 0.48. The average time taken by the QA system is 77.17 s, while that of the baseline system is 74.78 s. The performance of each user is also shown in Fig. 4. The QA system outperforms the baseline system in terms of the overall score for 7 out of 10 users (except for users 5, 9, and 10).

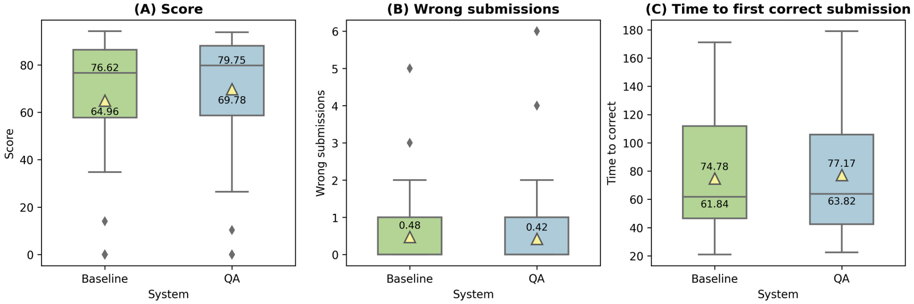

Fig. 3. (A) Overall score, (B) Time taken to answer the questions, and (C) Number of wrong submissions of the two systems.

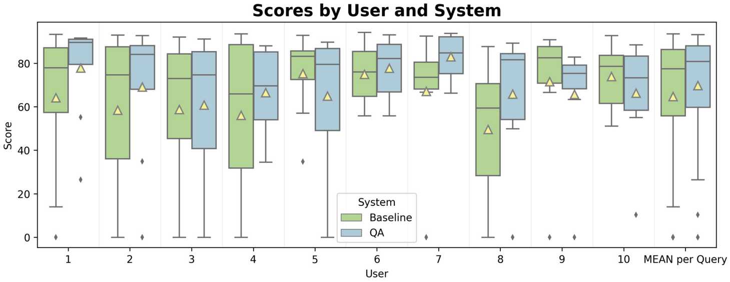

Fig. 4. Overall score of each user.

6.2 Importance of Experience

The results show that the QA system outperforms the baseline system in terms of the overall score. However, the performance of the QA system is not significantly better than the baseline system. This may be attributed to the fact that the participants had very little experience with lifelogging and question answering. To have a better understand of how the users perform with more experience, we analyse the average scores of the first system and the second system used by each user. The results are shown in Table 1. The average score of the first system used by each user is 66.67, while that of the second system used by each user is 68.06. This is expected as the users are more familiar with the tasks after using the first system. However, the average score of the first system used by the QA group (71.55) is higher than that of the baseline group (61.80). This indicates that the QA system is easier to use than the baseline system. Considering the second system only, the difference between two systems are not significant (68.12 for the baseline system and 68.00 for the QA system). The observed outcomes may be explained by the users getting familiar with the tasks after using the first system.

Table 1. Average scores for the first and second systems by each user.

System	Baseline	QA	Overall
First system only	61.80	**71.55**	66.67
Second system only	**68.12**	68.00	**68.06**

6.3 User Experience Questionnaire

Figure 5 displays the results of the User Experience Questionnaire. The questionnaire is designed to assess both pragmatic and hedonic aspects of system usability. The initial four questions measure the pragmatic quality of the system, focusing on its usefulness and efficiency. In contrast, the last four questions examine the hedonic quality, evaluating the system's overall pleasantness and user engagement. As shown in Fig. 5, the QA system outperforms the baseline system in all aspects in the questionnaire, with the larger difference observed in the pragmatic category, where the QA system shows an average advantage of 1.3 points compared to the baseline (1.7 vs. 0.4). This pronounced difference indicates that the QA system is more useful and efficient than the baseline system in the context of lifelog question answering tasks. The 0.83 points of difference in the hedonic category (1.5 vs. 0.67) also suggests that the QA system is more engaging and fun to use than the baseline system, which may be attributed to the QA system's intuitive and user-friendly nature, as discussed in the previous section.

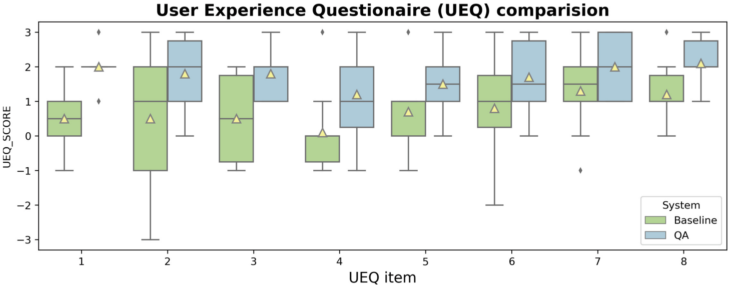

Fig. 5. Results of the User Experience Questionnaire.

7 Discussions and Conclusion

This paper presents a novel pipeline for integrating question answering capabilities into lifelog retrieval systems, which is based on the open-domain QA pipeline. By doing this, users can pose natural questions to the lifelogs and receive potential text answers. Our user study demonstrate the advantages of our QA system over the baseline approach in terms of overall scores and user satisfaction. Moreover, the results suggest that the QA system is a better option for new users.

In future works, deeper analysis on different question types is necessary to develop a well-rounded QA system. Furthermore, there are several ways to extend the QA pipeline, including result post-processing to improve the relevance of retrieved events, answer post-processing to re-rank the suggested answers, and answer highlighting to improve the user's confidence in the answers.

Acknowledgements. This work was conducted with the financial support of the Science Foundation Ireland Centre for Research Training in Digitally-Enhanced Reality (d-real) under Grant No. 18/CRT/6224. For the purpose of Open Access, the authors have applied a CC BY public copyright licence to any Author Accepted Manuscript version arising from this submission.

References

1. Alam, N., Graham, Y., Gurrin, C.: Memento 2.0: an improved lifelog search engine for LSC 2022. In: Proceedings of the 5th Annual on Lifelog Search Challenge, pp. 2–7 (2022)
2. Bain, M., Nagrani, A., Varol, G., Zisserman, A.: Frozen in time: a joint video and image encoder for end-to-end retrieval. In: IEEE International Conference on Computer Vision (2021)
3. Chang, C.C., Fu, M.H., Huang, H.H., Chen, H.H.: An interactive approach to integrating external textual knowledge for multimodal lifelog retrieval. In: Proceedings of the ACM Workshop on Lifelog Search Challenge, pp. 41–44 (2019)
4. Chen, D., Fisch, A., Weston, J., Bordes, A.: Reading wikipedia to answer open-domain questions. arXiv preprint arXiv:1704.00051 (2017)
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
6. Fabian Caba Heilbron, Victor Escorcia, B.G., Niebles, J.C.: ActivityNet: a large-scale video benchmark for human activity understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 961–970 (2015)
7. Gemmell, J., Bell, G., Lueder, R.: MyLifeBits: a personal database for everything. Commun. ACM **49**(1), 88–95 (2006)
8. Gurrin, C., et al.: Experiments in lifelog organisation and retrieval at NTCIR. In: Sakai, T., Oard, D.W., Kando, N. (eds.) Evaluating Information Retrieval and Access Tasks. TIRS, vol. 43, pp. 187–203. Springer, Singapore (2021). https://doi.org/10.1007/978-981-15-5554-1_13
9. Gurrin, C., et al.: Introduction to the fifth annual lifelog search challenge, LSC 2022. In: Proceedings of the International Conference on Multimedia Retrieval (ICMR 2022). ACM, Newark, NJ (2022)
10. Gurrin, C., et al.: Introduction to the sixth annual lifelog search challenge, LSC 2023. In: Proceedings of the International Conference on Multimedia Retrieval (ICMR 2023). ICMR 2023, New York (2023)
11. Izacard, G., Grave, E.: Leveraging passage retrieval with generative models for open domain question answering. arXiv preprint arXiv:2007.01282 (2020)
12. Karpukhin, V., et al.: Dense passage retrieval for open-domain question answering. arXiv preprint arXiv:2004.04906 (2020)

13. Laugwitz, B., Held, T., Schrepp, M.: Construction and evaluation of a user experience questionnaire. In: Holzinger, A. (ed.) USAB 2008. LNCS, vol. 5298, pp. 63–76. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-89350-9_6
14. Lee, J., Yun, S., Kim, H., Ko, M., Kang, J.: Ranking paragraphs for improving answer recall in open-domain question answering. arXiv preprint arXiv:1810.00494 (2018)
15. Lee, K., Chang, M.W., Toutanova, K.: Latent retrieval for weakly supervised open domain question answering. arXiv preprint arXiv:1906.00300 (2019)
16. Lewis, M., et al.: BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461 (2019)
17. Lewis, P., et al.: Retrieval-augmented generation for knowledge-intensive NLP tasks. Adv. Neural. Inf. Process. Syst. **33**, 9459–9474 (2020)
18. Liu, Y., et al.: RoBERTa: a robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
19. Nguyen, T.N., et al.: Lifeseeker 3.0: an interactive lifelog search engine for LSC 2021. In: Proceedings of the 4th Annual on Lifelog Search Challenge, pp. 41–46 (2021)
20. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning, pp. 8748–8763. PMLR (2021)
21. Raffel, C., et al.: Exploring the limits of transfer learning with a unified text-to-text transformer. J. Mach. Learn. Res. **21**(1), 5485–5551 (2020)
22. Rajpurkar, P., Jia, R., Liang, P.: Know what you don't know: unanswerable questions for squad. arXiv preprint arXiv:1806.03822 (2018)
23. Spiess, F., Schuldt, H.: Multimodal interactive lifelog retrieval with vitrivr-VR. In: Proceedings of the 5th Annual on Lifelog Search Challenge, pp. 38–42 (2022)
24. Tran, L.-D., Ho, T.C., Pham, L.A., Nguyen, B., Gurrin, C., Zhou, L.: LLQA - lifelog question answering dataset. In: Þór Jónsson, B., et al. (eds.) MMM 2022. LNCS, vol. 13141, pp. 217–228. Springer, Cham (2022). https://doi.org/10.1007/978-3-030-98358-1_18
25. Tran, L.D., Nguyen, B., Zhou, L., Gurrin, C.: Myeachtra: event-based interactive lifelog retrieval system for LSC 2023. In: Proceedings of the 6th Annual ACM Lifelog Search Challenge, pp. 24–29. Association for Computing Machinery, New York (2023)
26. Tran, L.D., et al.: Comparing interactive retrieval approaches at the lifelog search challenge 2021. IEEE Access **11**, 30982–30995 (2023)
27. Tran, L.D., Nguyen, M.D., Nguyen, B., Lee, H., Zhou, L., Gurrin, C.: E-myscéal: embedding-based interactive lifelog retrieval system for LSC 2022. In: Proceedings of the 5th Annual on Lifelog Search Challenge, pp. 32–37. LSC 2022, Association for Computing Machinery, New York (2022)
28. Tran, L.D., Nguyen, M.D., Nguyen, B.T., Zhou, L.: Myscéal: a deeper analysis of an interactive lifelog search engine. Multimedia Tools Appl. **82**, 1–18 (2023)
29. Tran, Q.L., Tran, L.D., Nguyen, B., Gurrin, C.: MemoriEase: an interactive lifelog retrieval system for LSC 2023. In: Proceedings of the 6th Annual ACM Lifelog Search Challenge, pp. 30–35 (2023)
30. Wang, S., et al.: R 3: reinforced ranker-reader for open-domain question answering. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32 (2018)
31. Yang, A., Miech, A., Sivic, J., Laptev, I., Schmid, C.: Zero-shot video question answering via frozen bidirectional language models. arXiv preprint arXiv:2206.08155 (2022)

32. Yang, W., et al.: End-to-end open-domain question answering with BERTserini. arXiv preprint arXiv:1902.01718 (2019)
33. Yu, J., Wang, Z., Vasudevan, V., Yeung, L., Seyedhosseini, M., Wu, Y.: Coca: contrastive captioners are image-text foundation models. arXiv preprint arXiv:2205.01917 (2022)
34. Zhou, L., et al.: Overview of the NTCIR-16 lifelog-4 task. In: Proceedings of the 16th NTCIR Conference on Evaluation of Information Access Technologies, pp. 130–135. National Institute of Informatics (2022)
35. Zhu, F., Lei, W., Wang, C., Zheng, J., Poria, S., Chua, T.S.: Retrieving and reading: a comprehensive survey on open-domain question answering. arXiv preprint arXiv:2101.00774 (2021)