

PUSHING THE BOUNDARIES OF LIFELOG RETRIEVAL SYSTEMS WITH QUESTION ANSWERING TECHNIQUES

Ly-Duyen Tran, B.Sc.

A Dissertation submitted in fulfillment of the
requirements for the award of
Doctor of Philosophy (PhD)

to the

DCU

Ollscoil Chathair
Bhaile Átha Cliath
Dublin City University

Dublin City University

Faculty of Engineering and Computing, School of Computing

Supervisors

Prof. Cathal Gurrin

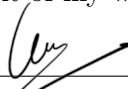
Dr. Liting Zhou

Prof. Owen Conlan, Trinity College Dublin

May 2024

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy (Ph.D.) is entirely my own work, and that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work

Sign:  _____
Ly-Duyen Tran

ID No.: 19213867

Date: 30/04/2024

Acknowledgements

I express my sincere gratitude to my supervisors, Prof. Cathal Gurrin, and Dr. Liting Zhou, and external supervisor Prof. Binh Nguyen (Ho Chi Minh City University of Science, Vietnam), for their valuable guidance and support throughout my Ph.D. I am grateful to my supervisors for their patience, motivation, enthusiasm, and immense knowledge. Their guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my Ph.D. study.

I would like to extend my appreciation to colleagues, friends, and fellow students who have encouraged my research. Especially, I would like to thank my colleagues in the Human Modelling Group for your shared experiences and discussions, as well as your support on technical skills, annotation, setting up experiments, and recruiting participants.

To my family, I cannot thank you enough for your love, support, and encouragement throughout my life. I am also grateful for my boyfriend's support and understanding during the last few months of my Ph.D. You have been a great motivation for me to complete this Ph.D.

Thanks to the main financial support from Science Foundation Ireland for the research in this thesis. The work of this PhD research has emanated from research conducted with the financial support of the SFI under grant agreement 13/RC/2106_P2 and the Centre for Research Training in Digitally-Enhanced Reality (d-real) under Grant No. 18/CRT/6224.

List of Publications

- [1] Han K. Cao, Duyen T. Ly, Duy M. Nguyen, and Binh T. Nguyen. “Automatically Generate Hymns Using Variational Attention Models”. In: *Lecture Notes in Computer Science*. Springer International Publishing, 2019, pp. 317–327.
- [2] Silvan Heller et al. “Interactive video retrieval evaluation at a distance: comparing sixteen interactive video search systems in a remote setting at the 10th Video Browser Showdown”. In: *International Journal of Multimedia Information Retrieval* 11.1 (2022), pp. 1–18.
- [3] Tu-Khiem Le, Manh-Duy Nguyen, Ly-Duyen Tran, Van-Tu Ninh, Cathal Gurrin, and Graham Healy. “DCU team at the NTCIR-15 Micro-activity Retrieval Task”. In: *Proceedings of the NTCIR-15 Conference*. 2020.
- [4] Thao-Nhu Nguyen et al. “DCU and HCMUS at NTCIR-16 Lifelog-4”. In: *Proceedings of the 16th NTCIR Conference on Evaluation of Information Access Technologies*. NTCIR. 2022.
- [5] Teketo Kassaw Tegegne, Ly-Duyen Tran, Rebecca Nourse, Cathal Gurrin, and Ralph Maddison. “Daily Activity Lifelogs of People With Heart Failure: Observational Study”. In: *JMIR Formative Research* 8 (Feb. 2024), e51248.
- [6] Ly-Duyen Tran, Cathal Gurrin, and Alan F. Smeaton. “Lifelogging As An Extreme Form of Personal Information Management – What Lessons To Learn”. In: (Jan. 11, 2024).
- [7] Ly-Duyen Tran, Thanh Cong Ho, Lan Anh Pham, Binh Nguyen, Cathal Gurrin, and Liting Zhou. “LLQA-Lifelog Question Answering Dataset”. In: *MultiMedia Modeling: 28th International Conference, MMM 2022, Phu Quoc, Vietnam, June 6–10, 2022, Proceedings, Part I*. Springer. 2022, pp. 217–228.
- [8] Ly-Duyen Tran, Diarmuid Kennedy, Liting Zhou, Binh Nguyen, and Cathal Gurrin. “A virtual reality reminiscence interface for personal lifelogs”. In: *International Conference on Multimedia Modeling*. Springer. 2022, pp. 479–484.
- [9] Ly-Duyen Tran, Manh-Duy Nguyen, Nguyen Thanh Binh, Hyowon Lee, and Cathal Gurrin. “Myscéal: An Experimental Interactive Lifelog Retrieval System for LSC’20”. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*. LSC ’20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 23–28.
- [10] Ly-Duyen Tran, Manh-Duy Nguyen, Nguyen Thanh Binh, Hyowon Lee, and Cathal Gurrin. “Myscéal 2.0: A Revised Experimental Interactive Lifelog Retrieval System for LSC’21”. In: *Proceedings of the 4th Annual on Lifelog Search Challenge*. 2021, pp. 11–16.

- [11] Ly-Duyen Tran, Manh-Duy Nguyen, Binh Nguyen, Hyowon Lee, Liting Zhou, and Cathal Gurrin. “E-Myscéal: Embedding-Based Interactive Lifelog Retrieval System for LSC’22”. In: *Proceedings of the 5th Annual on Lifelog Search Challenge*. LSC ’22. Newark, NJ, USA: Association for Computing Machinery, 2022, pp. 32–37.
- [12] Ly-Duyen Tran, Manh-Duy Nguyen, Binh T Nguyen, and Cathal Gurrin. “An Experiment in Interactive Retrieval for the Lifelog Moment Retrieval Task at ImageCLEFlifelog2020”. In: *CLEF (Working Notes)*. 2020, p. 12.
- [13] Ly-Duyen Tran, Manh-Duy Nguyen, Binh T Nguyen, and Liting Zhou. “Myscéal: a deeper analysis of an interactive lifelog search engine”. In: *Multimedia Tools and Applications* (2023), pp. 1–18.
- [14] Ly-Duyen Tran, Dongyun Nie, Liting Zhou, Binh Nguyen, and Cathal Gurrin. “VAISL: Visual-aware identification of semantic locations in lifelog”. In: *International Conference on Multimedia Modeling*. Springer. 2023, pp. 659–670.
- [15] Ly-Duyen Tran, Liting Zhou, Binh Nguyen, and Cathal Gurrin. “Interactive Question Answering for Multimodal Lifelog Retrieval”. In: *Lecture Notes in Computer Science*. Springer Nature Switzerland, 2024, pp. 68–81.
- [16] Ly-Duyen Tran et al. “A VR Interface for Browsing Visual Spaces at VBS2021”. In: *MultiMedia Modeling*. Ed. by Jakub Lokoc et al. Vol. 12573. Cham: Springer International Publishing, 2021, pp. 490–495.
- [17] Ly-Duyen Tran et al. “An Exploration into the Benefits of the CLIP model for Lifelog Retrieval”. In: *Proceedings of the 19th International Conference on Content-based Multimedia Indexing*. 2022, pp. 15–22.
- [18] Ly-Duyen Tran et al. “Comparing Interactive Retrieval Approaches at the Lifelog Search Challenge 2021”. In: *IEEE Access* 11 (2023), pp. 30982–30995.
- [19] Ly Duyen Tran, Binh Nguyen, Liting Zhou, and Cathal Gurrin. “MyEachtra: Event-Based Interactive Lifelog Retrieval System for LSC’23”. In: *Proceedings of the 6th Annual ACM Lifelog Search Challenge*. 2023, pp. 24–29.
- [20] Quang-Linh Tran, Ly-Duyen Tran, Binh Nguyen, and Cathal Gurrin. “MemoriEase: An Interactive Lifelog Retrieval System for LSC’23”. In: *Proceedings of the 6th Annual ACM Lifelog Search Challenge*. 2023, pp. 30–35.
- [21] Liting Zhou et al. “Overview of the NTCIR-17 Lifelog-5 Task”. In: *Proceedings of the 17th NTCIR Conference on Evaluation of Information Access Technologies (NTCIR-17)*. Tokyo, Japan, Dec. 2023.

Contents

List of Publications	v
1 Introduction	1
1.1 Background	1
1.1.1 Lifelog: A Digital Memory	1
1.1.2 Multimedia Retrieval Systems	3
1.2 Motivation and Research Problem	5
1.3 Hypothesis and Research Questions	6
1.4 Significance of the Research	8
1.5 Limitations	9
1.6 Research Contribution	10
1.7 Dissertation Outline	11
2 Literature Review	13
2.1 Lifelog Retrieval	13
2.1.1 Lifelog Data	14
2.1.2 Early Lifelog Retrieval Systems	15
2.1.3 Benchmarking Challenges	20
2.1.4 Approaches	22
2.1.5 Discussion	35
2.2 Approaches to Lifelog Question Answering	36
2.2.1 Open-Domain Question Answering (OpenQA)	38
2.2.2 Comprehension Question Answering	39
2.2.3 Relating to Lifelog Question Answering	42
2.3 Conclusion	44
3 Methodology	45
3.1 Design Science Research	46
3.2 Research Design	47
3.2.1 Research Objectives	47
3.2.2 Data	49
3.2.3 Ethical Considerations of Lifelog Data	53
3.2.4 Evaluation Criteria	53
3.2.5 Users	56
3.3 Operating Constraints	57
3.4 Conclusion	58

4	Interactive Lifelog Retrieval	59
4.1	Data Processing	61
4.1.1	Visual descriptors	61
4.1.2	Non-visual metadata processing	62
4.1.3	Temporal units	64
4.2	Search Process	64
4.2.1	ElasticSearch indexing	65
4.2.2	Query parsing	65
4.2.3	Primary search mechanism	66
4.2.4	Complementary search mechanisms	67
4.3	User Interaction	68
4.3.1	Search boxes	69
4.3.2	Query suggestion	69
4.3.3	Search results display	70
4.3.4	Map	71
4.3.5	Visual Similarity and Event View	71
4.4	Performance	73
4.4.1	Myscéal at LSC'20	73
4.4.2	Myscéal-CLEF at ImageCLEFlifelog'20	76
4.4.3	Myscéal 2.0 at LSC'21	78
4.4.4	E-Myscéal at LSC'22	80
4.5	Discussion	81
4.6	Conclusion	84
5	Contextual Lifelog Question Answering	85
5.1	Task Definition and Dataset Requirements	85
5.2	Dataset Construction	87
5.2.1	Description Collection	87
5.2.2	Generate Question and Answers	88
5.2.3	Review	89
5.3	Dataset Analysis	90
5.3.1	Dataset Limitations	91
5.4	Evaluation	92
5.4.1	Baselines	93
5.4.2	Pretrained Video-Language Models	94
5.4.3	Benchmarking Results	101
5.5	Discussion	103
5.6	Conclusion	104
6	Event-based Embeddings	107
6.1	MyEachtra	108
6.1.1	Event-Based Approach	108
6.1.2	Displaying Events	109
6.2	Evaluation Using LSC'22 Queries	110
6.3	User Study: Comparison with Myscéal	112
6.4	MyEachtra at LSC'23	114
6.5	Discussion	115

6.6	Conclusion	118
7	Lifelog Question Answering System	119
7.1	Lifelog Question Answering Pipeline	119
7.1.1	Event Retriever	120
7.1.2	Event Reader	121
7.2	User Study Setup	122
7.2.1	Lifelog Questions	122
7.2.2	User Study Design	124
7.2.3	Baseline System	125
7.2.4	QA system	125
7.3	User Study Results	126
7.3.1	Overall Score	126
7.3.2	Importance of Experience	127
7.3.3	User Experience Questionnaire	128
7.4	MyEachtra In LSC'23	129
7.5	Discussion	131
7.6	Conclusion	132
8	Conclusion	135
8.1	Hypothesis and Research Questions	135
8.2	Research Contributions	138
8.3	Limitations	138
8.4	Future Works	140
A	List of Lifelog Queries Used in Lifelog Experiments	143
	References	155

List of Abbreviations

BERT Bidirectional Encoder Representations from Transformers.

CLIP Contrastive Language-Image Pre-training.

CNN Convolutional Neural Network.

DCU Dublin City University.

DSR Design Science Research.

GPT Generative Pre-trained Transformer.

GRU Gated Recurrent Unit.

IR Information Retrieval.

KIS Known Item Search.

LLQA Lifelog Question Answering.

LMRT Lifelog Moment Retrieval Task.

LSC Lifelog Search Challenge.

LSTM Long Short-Term Memory.

MLM Masked Language Model.

MRC Machine Reading Comprehension.

NLP Natural Language Processing.

OCR Optical Character Recognition.

OpenQA Open-domain Question Answering.

QA Question Answering.

RAG Retrieval Augmented Generation.

RNN Recurrent Neural Network.

ST Syntax Transformation.

UEQ User Experience Questionnaire.

VBS Video Browser Showdown.

VTM Visual Textual Model.

List of Tables

2.1	An example KIS task from LSC'20 [212]. Task 1 with its temporally advancing descriptors, which were revealed at 30-second intervals. After 150 seconds, the full description is shown for another 150 seconds until the end of the task.	22
2.2	Selected approaches used by participating systems, adapted from [212]. For each system, a reference to the paper describing the method is given. The order of the systems is based on their ranking each year. The systems in bold are the ones that I have worked on.	36
3.1	Design Science Research Guidelines, borrowed from [79]	46
3.2	Selected non-visual metadata from both datasets. Visual concepts are not shown here due to space limitations and the fact that this dissertation did not make use of the provided visual concepts, but rather employed my own visual concept detection models.	52
3.3	Precision and Recall calculation	55
4.1	Comparison of different versions of Myscéal.	60
4.2	ElasticSearch document for each scene.	65
4.3	Properties of ElasticSearch document for each image. For visual concepts that lack areas, we use the ElasticSearch keyword data type and configure ElasticSearch to calculate the TF-IDF scores. *: only available in E-Myscéal.	66
4.4	List of tasks used in our novice user study. The tasks were chosen from the query bank used in LSC'20. The symbol '/' separates clues that are gradually revealed to searchers.	75
4.5	Experiment score of eight novice users compared to Myscéal team's official score in LSC'20.	76
4.6	$F1@10$ scores of three users (U1: Lifelogger, U2: Expert, U3: Novice). The symbol '-' indicates that the user was unable to find the answer for that task. The numbers with * are the highest number in that topic.	77
4.7	Summary of LSC'21 result of top-6 systems. The numbers in bold are the highest numbers among the top 6 systems. Precision and recall are defined in Section 4.4.1.	79
5.1	Numbers of questions in each month in LSC'20 lifelog data collection.	90
5.2	Accuracy (%) of different models in the pilot experiment.	95

5.3	Results of more recent SOTA models in video QA on LLQA dataset. All models were evaluated on both yes/no and multiple-choice questions at the same time (Overall). To indicate the use of metadata, (+m) is added to the model name. WV stands for WebVid.	102
6.1	Mean $H@K$ for LSC'22 queries using Mean Pooling. I am most interested in the modified version of H@3 because (i) once the user find the correct answer, more hints are not needed and (ii) the user interface can display three events at a time.	112
6.2	Latin Square design for the user study to evaluate the event-based MyEachtra system on LSC'22 KIS queries. A and B represent MyEachtra and Myscéal respectively. Q1–Q8 represents the eight queries.	113
6.3	Statistics of each system's performance in the user study for KIS queries in LSC'22.	113
7.1	Average score of the first system and the second system used by each user. .	128
A.11	User Experience Questionnaire	154

List of Figures

2.1	SenseCam Photo Viewer interface as reported in [81].	16
2.2	MyLifeBits query result interface.	17
2.3	MyLifeBits' map view [13]. The map on the right shows large dots where photos are taken and small dots for GPS track points.	18
2.4	SenseCam Visual Diary interface as reported in [112].	19
2.5	Some lifelog images as ground truth from Task I in Table 2.1	21
2.6	The general pipeline of a lifelog retrieval system.	23
2.7	Faceted filters in lifeXplore [147]	29
2.8	Relevance feedback in Exquisitor [99]	30
2.9	Sketch-based search in vitrivr [75]	31
2.10	Results in triplets in MemoriEase [216].	33
2.11	Data filtering visualisation in Memento [4]	33
2.12	Location transitional graph with fine-grained location hierarchy in Life-Seeker [110].	34
2.13	Virtual Reality Interface of UU-DCU team in LSC'18.[46].	35
2.14	The architecture of DrQA [29], a typical IR-based QA system consisting of a Retriever and a Reader.	39
3.1	Timeline of the research cycles	47
3.2	Example lifelog images from both datasets.	51
4.1	Pipeline of MyScéal.	59
4.2	GPS clustering in Myscéal 2.0.	63
4.3	User interface of Myscéal. The search can be initiated by filling in the search boxes (A) and pressing the <i>Enter</i> key on the keyboard. The search results are shown on the left side of the screen (B) in a list of triplets. The map on the right side of the screen (C) shows the locations of the images in the search results. The bottom right corner of the screen shows the saved section (D), which allows the user to save the search results for later use.	69
4.4	Event View window in Myscéal, Myscéal-CLEF, and Myscéal-2.0.	70
4.5	User interface of Myscéal in ImageCLEF 2020	71
4.6	Event View window in E-Myscéal, merged with the Visual Similarity view.	72
4.7	Precision and Recall of each team in LSC'20.	73
4.8	Number of times each team was in the top 3 quickest teams to return the correct results.	74
4.9	Overall score of all teams in LSC'22.	80
4.10	Number of incorrect and correct submissions of E-Myscéal in LSC'22.	81

4.11	Time to first correct submission of different teams in LSC'22. Ad-hoc tasks are not included as they are not scored based on time.	82
4.12	Precision and Recall of the Ad-hoc tasks in LSC'22.	83
5.1	The process of dataset construction.	87
5.2	Annotation Interface	88
5.3	The procedure of question-answer generation.	88
5.4	Two example question-answer pairs in the dataset. The dataset contains both yes/no questions and multiple-choice questions.	89
5.5	Numbers of each question type in Contextual lifelog QA dataset.	90
5.6	Distribution of the first four words in the questions.	91
5.7	Getting the CLIP embeddings. The weights of the CLIP model are frozen (shown in blue) during training.	98
5.8	MeanQA model.	99
5.9	SelfQA model.	100
5.10	CrossQA model.	101
5.11	FullCrossQA model.	101
6.1	MyEachtra's user interface. Each row represents an event. The most relevant image is highlighted and placed in the middle of the row.	108
6.2	Hit Rate at K at different hints on four approaches.	111
6.3	Comparison between MyEachtra and E-Myscéal for LSC'22 queries.	114
6.4	Overall score of all teams in LSC'23. The baseline system is E-Myscéal.	114
6.5	Number of incorrect and correct submissions of teams in LSC'23.	115
6.6	Time to first correct submission all teams in LSC'23 in KIS tasks. Ad-hoc tasks are not included because the time does not matter in this task.	116
6.7	Precision and Recall for Ad-hoc tasks of all teams in LSC'23.	117
7.1	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.	120
7.2	Distribution of the questions in the test collection.	124
7.3	MyEachtra's user interface. For non-QA tasks, the left panel is hidden.	126
7.4	(A) Overall score and (B) Time taken to answer the questions of the two systems.	127
7.5	Overall score of each user.	127
7.6	Average scores across different types of questions.	128
7.7	Results of the User Experience Questionnaire. MyEachtra significantly outperforms the baseline system in all aspects (p-value is nearly 0).	129
7.8	Overall score of all teams in LSC'23 for QA tasks. MyEachtra achieved the highest score. The baseline system, E-Myscéal, achieved the fifth spot in the ranking with 71% of the score of MyEachtra (p-value = 0.0007 for the average score per question).	130
7.9	Accuracy of submissions of all teams in LSC'23.	130
7.10	Time to first correct submission all teams in LSC'23 in QA tasks.	131

Pushing the Boundaries of Lifelog Retrieval Systems with Question Answering Techniques

Ly-Duyen Tran

Abstract

Lifelogging, referring to the continuous capturing of personal experiences using digital devices such as wearable cameras and smart sensors, could be a valuable memory enhancement and provide great insights into how an individual lives their life. This thesis focuses on the novel application of question answering (QA) within the context of lifelogging, aiming to develop an advanced interactive lifelog retrieval system that supports answering questions based on lifelog data. To achieve this objective, this research addresses several key components. First, a novel lifelog QA dataset, named LLQA, was created, consisting of over 15,000 multiple-choice and yes-no questions regarding data from lifelog segmentations. I evaluated different existing QA models on their suitability and capability to answer lifelog questions and compared their accuracies to the human baseline of 85.46%. The evaluation results recognised the efficiency of leveraging large pre-trained video-language models, achieving an accuracy of 72.43%, as opposed to constructing custom-built LLQA models, which achieved 71.23%. Next, I designed and continually enhanced MyScéal, a state-of-the-art interactive lifelog retrieval system that supports the user to efficiently retrieve relevant data in response to search queries and lifelog questions. This effort culminated in MyScéal's success as the winning system in three consecutive iterations of Lifelog Search Challenges, underscoring its strengths in supporting the user to quickly locate items of interest from a conventional multimodal lifelog. Finally, a novel lifelog QA pipeline was proposed to seamlessly integrate QA models into existing lifelog retrieval systems. To demonstrate the effectiveness of the proposed pipeline, I integrated a lifelog QA model into MyScéal with modifications and developed a dedicated lifelog QA system known as MyEachtra. User studies were carried out to analyse the strengths and weaknesses of MyEachtra. The results showed that MyEachtra effectively supports the user in answering lifelog questions and enhances overall user satisfaction. The findings of this research have the potential to establish a foundation for further exploration into the task of lifelog QA.

Chapter 1

Introduction

Thanks to advances in wearable devices and mobile technologies, lifelogging has become a popular topic in recent years [64]. In order to manage the large amount of data collected, lifelog retrieval systems are tools that allow users to access relevant information from lifelogs. However, there has been a lack of research in applying Question Answering (QA) approaches into lifelog retrieval systems to support natural questions that one might have about one’s past experiences. This research aims to address this gap by proposing and evaluating a pipeline for lifelog QA and incorporating existing QA techniques into a lifelog retrieval system. This chapter will provide an introduction to lifelog and lifelog retrieval systems, followed by introducing the research problem, the research questions, the significance of the research, the limitations, and finally, the organisation of the dissertation.

1.1 Background

1.1.1 Lifelog: A Digital Memory

What if we could keep our life experiences so that they are never lost? What if we could get the answer to any question about our past? We might worry less about forgetting where we have left our keys or some important documents we need for a meeting today and focus more on living in the moment. The embarrassment of forgetting someone’s name might also be avoided. What about providing accurate answers to the doctor’s questions in our next appointment as how often we take the stairs instead of the lift? This exciting vision of ‘digital memory’ is one of the motivations for many researchers to study **lifelogging** — the process of capturing and storing a large amount of data about one’s life experiences [64]. Such collections of data, referred to as **lifelogs**, encompass a diversity of data types, for instance, images, videos, audio recordings, textual annotations, location information, and sensor readings, aiming to document *as much as possible* about an individual’s life. A variety of devices are used to capture lifelog data, including wearable cameras, GPS trackers, and biometrics sensors, resulting in a rich and diverse multimodal

data archive that can be used to answer questions about a person's past. Started from the idea of Vannevar Bush's 'Memex', a personal machine that can manage all your documents and provide quick access to any information that it contains, what was once considered a futuristic vision has become feasible due to today's advances in wearable computing and affordable digital storage capacity.

In his 1945 article 'As We May Think' [24], Vannevar Bush described a blueprint personal information system which he called 'Memex'. Memex was envisioned as 'a device in which an individual stores all his books, records, and communications, which is mechanised to be consulted with exceeding speed and flexibility'. The concept of Memex was a major influence in the early development of hypertext[148], which is a precursor in the subsequent creation of the World Wide Web (WWW). Bush considered Memex as 'an enlarged intimate supplement to [one's] memory', which hints at parts of the idea of what we now call lifelogging. Despite the fact that Memex was never built, it has inspired many researchers to explore the idea of lifelogging.

Early research in lifelogging started with a focus on sensing technologies in order to capture the user's life experiences. In the 1980s, Steve Mann, often referred to as the father of wearable computing, contributed to the field with many generations of wearable cameras and pointed out fundamental challenges in developing such technologies[144]. The term 'lifelogging' was also coined by Mann with the foundation of a community of lifeloggers.

Another pioneer in the field is Gordon Bell, who was part of a project called MyLifeBits [59] at Microsoft Research Labs. The project aimed to fulfill Bush's conceptual vision of Memex by capturing every possible aspect of the daily life of Bell, including every web page visited, all Instant Message chat sessions, all telephone conversations, meetings, radio, television programs, as well as all mouse and keyboard activities and media files in his personal computers. All digitised data is stored in a SQL database to support a simple interface for different functionalities such as organising, associating metadata, assessing, and reporting information. Full-text search, text, and audio annotations, and hyperlinks are also supported by the system. MyLifeBits has been considered one of the first influential lifelogging systems and has inspired many research projects in the field.

Inspired by Gordon Bell's project, Cathal Gurrin has been gathering a detailed and extensive archive of lifelog data using a SenseCam since 2006, which is the largest longitudinal lifelog archive to date to the best of my knowledge. Sensor data such as locations, movements, and biometrics are also collected to provide a rich multimodal collection of data. The archive has been considered a valuable source for many research efforts to gain an understanding of the challenges in lifelogging and the potential of lifelogging technologies [64].

The applications of lifelogging are endless and have been explored in many domains, such as supporting human memory recollection [19, 21, 72], supporting large-scale epidemiological studies in healthcare [195], monitoring lifestyle of individuals [153, 222], behaviour

analytics [49], diet/obesity analytics [242]. However, the vast volume and immense complexity of lifelog archives present a challenge for users to navigate and analyse relevant information. As such, there is a growing interest in developing lifelog retrieval systems that effectively leverage lifelog data to meet the diverse needs of users.

1.1.2 Multimedia Retrieval Systems

Retrieval systems for lifelog data have been a popular research topic in the past few years. It is a crucial task in order to manage and make use of the large amount of data collected by lifeloggers. In MyLifeBits, state-of-the-art techniques from database search and traditional Information Retrieval (IR) were employed to index and provide access to lifelog data through a desktop interface. However, as the volume of lifelog data increases, the need for more efficient and effective retrieval systems has become more apparent. The first retrieval system that was designed for large archives of lifelog data was proposed by Doherty et al.[43], moving the time/date *browsing* approach of lifelog systems at the time to a *search* approach. The system employed event segmentation, event annotation and multi-axes search, which are the ‘who’, ‘what’, ‘when’, and ‘where’ axes of retrieval. However, without a large user base, it was difficult to define search use cases in order to evaluate and improve lifelog retrieval systems.

As such, Sellen and Whittaker[189] have proposed a set of motivations for accessing past memories, which can be used as a basis for the development of retrieval models. The motivations are known as the Five Rs: recollecting, reminiscing, retrieving, reflecting, and remembering intentions. The principles for designing lifelog systems based on the Five Rs are also proposed by Gurrin et al. in their book ‘Lifelogging: Personal Big Data’[64] as follows:

- *Recollecting* refers to the act of recalling specific past events or experiences to relive certain moments or retrieve particular information from these events. The systems that support recollecting should be able to accurately rank content and retrieve the most relevant sequence of lifelog data to the user’s information need, in as much detail as possible to help the user recollect the event. This requires conventional information retrieval, adaptive event segmentation, and appropriate presentation representation. Most of the existing lifelog retrieval systems are designed for this purpose, including the system proposed by Doherty et al.[43] mentioned above.
- *Reminiscing* refers to a special form of recollection for emotional or sentimental reasons such as sharing memories with others. The systems that support reminiscing should highlight visual modalities, integrate retrieval, and provide the ability to organise, generate narratives, detect novelty, and summarise the retrieved content. One example of a system that supports storytelling is the work of Byrne et al.[25].

- *Retrieving* refers to the act of retrieving specific information from the lifelog archive, such as a location, date, etc. Such retrieval requires higher precision than general recollecting, and most likely some inference is required to extract the information from the underlying data. The authors suggested that the retrieval mechanism should be similar to question-answering systems than to whole document retrieval.
- *Reflecting* refers to the act of analysing the lifelog archive to gain insights and knowledge that may not be immediately obvious. Such systems should be able to support data analysis and visualisation, and provide the ability to detect events, trends, and patterns. The system should also be able to support the user in constructing queries at query time, as it is difficult to pre-identify all types of reflection that could form user queries.
- *Remembering intentions* refers to the act of planning future activities. The system could remind people about tasks they would like to do or give prompts on real-time situational needs such as who they are talking to or what topics they could talk about based on past experiences. Most lifelogging research has been focused on, and there is little work on remembering intentions.

Although there is a considerable overlap between lifelog retrieval and conventional information retrieval, it is important to note that lifelog retrieval presents unique challenges because the fact that the main motivation for a user to use a lifelog retrieval system is that he or she has forgotten the details of the past events in the first place. This means that the information need of a user may not be well-defined, and the query may be vague, incomplete, or even incorrect. However, there has been little research on unambiguous query formulation for lifelog retrieval.

Several benchmarking platforms for lifelog retrieval systems have been organised, with the first ones being NTCIR [63] in 2016, Lifelog Moment Retrieval Task (LMRT) in ImageCLEF[33] in 2017, and the Lifelog Search Challenge (LSC)[66] in 2018. Both automatic and interactive lifelog systems have been evaluated in these platforms using various retrieval metrics. However, interactivity offers a more natural way for users to interact with lifelog data, addresses the ambiguity of queries as mentioned, and allows users to refine their queries based on the retrieved results. Thus, interactive lifelog retrieval systems have been the focus of many research efforts in recent years with the LSC being the most influential platform for such systems. The LSC has been organised annually since 2018, with the most recent one being LSC'23[71]. With a focus on interactivity and user experience, the LSC has been a valuable platform for the development of lifelog retrieval systems that are accessible. 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 (for example 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[209], BM25[27, 216], bag-of-words (BoW)[150] 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[199] or map visualisation[209]. Recently, with the rise of cross-modal embedding models, such as CLIP[167] and CoCa[235], 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[4, 207]. 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. My study aims to contribute to this area by proposing a pipeline for integrating QA capabilities into lifelog systems and evaluating its effectiveness compared to baseline search-only lifelog systems.

1.2 Motivation and Research Problem

The objective of lifelog retrieval systems is to facilitate efficient access to lifelog data, allowing users to search and browse through lifelogs to retrieve relevant information. Because of the multimodal nature of lifelog archives, these systems often apply state-of-the-art techniques from various domains, notably multimedia retrieval, image processing, and computer vision to organise, segment, and annotate lifelog data[41, 67, 212].

However, despite the advancements in lifelog retrieval systems, a significant limitation persists: it is human nature to ask *questions* about our past experiences, and in fact, we often do so in our daily lives. For example, we may ask ourselves ‘Where did I leave my keys?’ or ‘When did I last meet with my friend?’. This use-case is aligned with the *Retrieving* motivation of lifelog retrieval systems discussed in the last section. Most existing lifelog retrieval systems focus on the search task, where the system ranks the relevant lifelog images based on the user’s query (*Recollecting* motivation). While these systems can be used to answer questions, they require manual browsing through the ranked images and inferring the answer from them. This is not only time-consuming but also requires the user to have a good understanding of the system and the underlying data. Furthermore, most of these systems are usually designed for expert users, who are familiar

with the concept of lifelogging, the data structure, and the retrieval system itself. Novice users, who are new to lifelogging and have little to no experience with such systems may find it overwhelming to interact with the system and find the answer they need.

Consequently, there is a need for a lifelog retrieval system that addresses question answering (QA) tasks directly, enabling users to ask specific questions about their lifelogs and receive text-based answers. This integration of QA functionalities will not only enhance the user experience but also make lifelogging more valuable and accessible for novice users. While the potential of QA has been recognised in various domains, there has been little research on applying QA to lifelog.

The central focus of this dissertation is to address the steps required to design a state-of-the-art interactive lifelog retrieval system that (1) assists the novice user to quickly locate items of interest from a conventional multimodal lifelog, and (2) incorporates tailored approaches to lifelog question answering to improve the user experience and overall performance of interactive lifelog retrieval tasks. By addressing the steps required to design such a system, this research aims to make lifelog retrieval more accessible for a broader user base, thereby supporting the potential applications of lifelogging in various domains.

1.3 Hypothesis and Research Questions

In order to achieve the research goal of designing a state-of-the-art interactive lifelog retrieval system with QA capabilities, this dissertation aims to prove or disprove the following hypothesis:

Hypothesis *Question Answering techniques can improve upon state-of-the-art interactive retrieval systems for lifelog data by improving the result's quality and supporting quick access to relevant information.*

Several related research questions have been developed to guide the research process, as follows:

Research Question 1 (RQ1). **How to design a state-of-the-art interactive lifelog retrieval system that assists a novice user to quickly locate items of interest from a conventional multimodal lifelog?**

First of all, it is necessary to develop a state-of-the-art interactive lifelog retrieval system that can be used as a baseline for the incorporation of QA techniques. This system should be able to effectively retrieve relevant lifelog data in response to search queries by leveraging state-of-the-art techniques from various domains and following the best practices in lifelog retrieval. Additionally, the system should be designed with novice users in mind, with a simple and intuitive interface that is easy to use. To answer this question, I will develop a novel interactive lifelog retrieval system, Myscéal, and evaluate

its performance against other lifelog systems in benchmarking activities to demonstrate that it is a state-of-the-art system.

Myscéal contributes directly to the field of lifelog retrieval by advancing the state-of-the-art and emphasising the choice of simplicity over complexity in the design of lifelog retrieval systems. Addressing this question is crucial for improving the usability and accessibility of lifelog systems, making lifelogging more inclusive and appealing to a broader audience.

Research Question 2 (RQ2). How can we evaluate different approaches to question answering on lifelog datasets?

While QA has been intensively studied in other domains, it remains an underexplored area in the field of lifelogging. This research question focuses on the first step of incorporating QA techniques into lifelog retrieval systems. Specifically, I plan to answer the following sub-questions:

RQ2.1. How to adapt existing lifelog test collections to evaluate approaches to lifelog question answering? This research question focuses on the development of a lifelog QA dataset, which is necessary to evaluate the effectiveness of various QA approaches. To answer this question, I will construct a lifelog QA dataset to support the evaluation of lifelog QA techniques. The existing lifelog collection from Lifelog Search Challenge (LSC)[68] is a valuable source of data and should be used as a basis for the development of the lifelog QA dataset. The dataset should be annotated with questions that are relevant to the lifelog data in the form of multiple-choice or yes-no questions. In order to reduce the annotation cost, I will collect lifelog captions from multiple annotators and employ a set of techniques to automatically generate questions from these captions.

RQ2.2. What existing question answering techniques are most effective when applied to lifelog data? QA techniques have achieved significant success in other domains such as text QA, visual QA, and video QA. However, it is unclear which of these techniques are most effective when applied to lifelog data. To answer this question, I will evaluate the effectiveness of various QA approaches on the LLQA dataset. The approaches include pretrained zero-shot models, models fine-tuned on the LLQA dataset, and hybrid models that build on top of frozen pretrained models. According to the result, the most suitable QA techniques for lifelog will be identified and used in the development of a dedicated lifelog QA system.

Research Question 3 (RQ3). Can incorporating tailored approaches to lifelog question answering result in improved novice user performance on interactive lifelog retrieval tasks, when compared to existing state-of-the-art interactive lifelog retrieval systems?

This research question focuses on the incorporation of tailored question answering approaches into interactive lifelog retrieval tasks by evaluating the system’s performance and comparing it with the existing state-of-the-art lifelog retrieval system (Myscéal). To answer this question, I propose two sub-questions:

RQ3.1. Does the event-based retrieval support the user to achieve comparative performance to image-based retrieval for lifelog data? To accommodate lifelog QA approaches to lifelog retrieval systems, the search results must be in a format similar to the LLQA dataset. This means that instead of retrieving individual images, the results must be presented in the form of ‘events’, which are continuous sequences of lifelog moments (images and the corresponding metadata). Thus, I will develop an event-based retrieval approach that groups lifelog images into events and retrieves relevant events directly instead of individual images. After that, I will evaluate the performance of the event-based retrieval approach by comparing it with the conventional image-based retrieval approach.

RQ3.2. Can the tailored question answering approach improve the performance of interactive lifelog retrieval? To answer this question, I will design a framework to incorporate the most suitable QA techniques for lifelog retrieval tasks. The framework should be able to support both conventional search and QA tasks, allowing users to choose between the two modes. A user study will be conducted to evaluate the performance of the tailored question answering approach in interactive lifelog retrieval tasks. Furthermore, a new test collection should be created for the user study, which focuses more on general, open-domain lifelog questions as opposed to the specific questions in the LLQA dataset. The results of the user studies will be used to evaluate the effectiveness of the tailored question answering approach and compare it with the existing state-of-the-art lifelog retrieval system (Myscéal).

Research Question 3 directly aligns with the overall research goal, which is to design a state-of-the-art interactive lifelog retrieval system with QA capabilities. By answering this question, I will demonstrate the effectiveness of our proposed pipeline for incorporating QA techniques into lifelog retrieval systems, thereby fulfilling the objective of this dissertation.

1.4 Significance of the Research

This research contributes to the body of knowledge in the field of lifelogging and raises awareness of the task of lifelog question answering for the multimedia analytics community

by addressing the steps required to design a state-of-the-art interactive lifelog retrieval system with QA capabilities. The proposed system sets new standards for lifelog retrieval and can be used as a baseline for future research. Moreover, it has the potential to extend the limited knowledge on the use cases of lifelogging and user interaction with lifelog data. In addition, by catering to a broader user base, the proposed system can help to make lifelogging more inclusive and appealing to a wider audience, thereby increasing the adoption of lifelogging technologies in various research domains such as human-computer interaction, information retrieval, natural language processing, and computer vision; as well as practical domains such as personal memory augmentation, healthcare, and lifestyle management.

1.5 Limitations

While this research aims to make significant contributions to the field of interactive lifelog retrieval, several limitations need to be acknowledged:

- **Data Diversity:** The proposed system is evaluated on a longitudinal, multimodal lifelog dataset that despite its large volume and high complexity, is still limited to one lifelogger. The system’s performance may vary significantly when applied to other lifeloggers’ data. However, at present, there are no other longitudinal and multimodal lifelogs from multiple lifeloggers.
- **User Diversity:** The proposed interactive lifelog retrieval system caters to novice users. However, the size of the user study is relatively small, and the participants are mostly students.
- **Limited Resources:** The proposed system is developed using one computer with limited resources. The system’s performance may be improved by using more powerful machines and more resources. This also applies to the development of the lifelog QA dataset, which is limited by the availability of annotators and the time required to annotate the data.
- **Data Privacy and Security:** In order to protect the privacy of people in the lifelog data and to meet the expectations of our institutional ethics committee, the data is anonymised. Therefore, lifelog queries and questions in this research ignore the human interaction aspect of lifelog data.
- **Evaluation Metrics:** The proposed system is evaluated using the scoring metrics of the Lifelog Search Challenge[68], which is the metric used in this community. Therefore, it is also used in this research to ensure that the proposed system is comparable with other systems. While these metrics are suitable for comparing the

performance of different lifelog retrieval systems, there may be better metrics for evaluating the effectiveness of lifelog question answering approaches. However, they are not the focus of this research.

- **Dependency on External Factors:** The system’s performance may be influenced by external factors such as the quality of natural language processing and computer vision tools, or the availability of pretrained models.
- **Multimedia Analytics:** This research is performed within the multimedia domain as an exercise in multimedia analytics rather than information retrieval. As a result, while there are numerous QA techniques that could be used, I will only cover multimedia topics in this thesis.

Synthetic lifelog data has been discussed, albeit informally, as a potential solution address the data sparsity, diversity, and privacy issues. This approach is widely used in the field of computer vision and natural language processing to evaluate the performance of various models[138]. However, there has been little research on the generation of synthetic lifelog data, and the effectiveness of this approach is still unclear.

Acknowledging these limitations is essential for ensuring a comprehensive understanding of the research scope and potential challenges. Despite these limitations, the research lays a strong foundation for future work in interactive lifelog retrieval and question answering, stimulating further investigations to address these challenges and advance the state-of-the-art in lifelogging technologies.

1.6 Research Contribution

The key contributions of this dissertation are as follows:

- A ranking algorithm for image-based retrieval, aTF-IDF, that is inspired by the traditional TF-IDF algorithm and is designed to be used in lifelog retrieval systems.
- An implementation of the temporal search functionality, which set a trend for lifelog retrieval systems to support temporal search in recent years.
- A lifelog QA dataset, LLQA, that is publicly available and also comes with a set of valuable lifelog captions.
- A set of experiments and analyses of various QA techniques on the LLQA dataset, including pretrained zero-shot models, models fine-tuned on the LLQA dataset, and hybrid models that build on top of frozen pretrained models.
- An event-based retrieval approach that groups lifelog images into events and retrieves relevant events directly instead of individual images. This is a novel approach that has not been explored in the field of lifelog.

- A framework for incorporating QA techniques into interactive lifelog retrieval systems, which is the most important contribution of this work. This can be used as a guideline for future research in this area.
- A second lifelog QA dataset that does not provide the context required to answer the questions, which is considered much more challenging than the LLQA dataset.

1.7 Dissertation Outline

In this dissertation, the research questions are addressed by developing a state-of-the-art interactive lifelog retrieval system and incorporating suitable QA techniques into the system. This introduction chapter provided insight into the motivation behind this research, the challenges faced, and the significance of the study. The research questions and the expected contributions of the research were also introduced. The remainder of this dissertation is organised as follows:

- Chapter 2 provides a comprehensive literature review that identifies state-of-the-art approaches in lifelog retrieval and QA, as well as their limitations. This lays the foundation for the proposed baseline and the proposed QA system.
- Chapter 3 describes the action research paradigm that I adopted for this dissertation with the operating constraints of this research. The chapter also covers the pipeline design for incorporating QA techniques into lifelog retrieval system. Furthermore, I explain the lifelog datasets used for evaluation, the live benchmarking challenges, user study setups, and the evaluation metrics for lifelog retrieval, especially lifelog QA, approaches.
- Chapter 4 presents the first contribution of this dissertation, which is a state-of-the-art interactive lifelog retrieval system, Myscéal. It describes different components of the system, including changes made throughout the development process, and the performance of the system in various lifelog retrieval challenges.
- Chapter 5 introduces the LLQA dataset, which is the first lifelog QA dataset that is publicly available. Various QA techniques on the LLQA dataset are also discussed in this chapter in order to identify the most suitable QA techniques for lifelog retrieval tasks.
- Chapter 6 describes the first modifications of the Myscéal system to incorporate the lifelog QA techniques, shifting the focus from image-based retrieval to event-based retrieval.
- Chapter 7 proposes a framework for incorporating QA techniques into interactive lifelog retrieval systems, which is the most important contribution of this work. This

framework is evaluated by comparing the performance of the proposed QA-support system with the baseline Myscéal system,

- Finally, Chapter 8 concludes the dissertation by summarising the contributions and findings of this research and discussing the potential areas for future research.

Chapter 2

Literature Review

As wearable technologies have become more and more sophisticated and affordable, lifelogging has become a popular topic in recent years. To manage the large volume of lifelog data, lifelog retrieval systems have been developed to assist users to organise, browse, and search through their lifelogs. However, question answering (QA) remains under explored in the context of lifelogging. In this chapter, I will review the literature on lifelog retrieval systems and QA techniques in various related domains to approach the research questions of this thesis. The different types of lifelog data, the earliest lifelog retrieval systems, benchmarking efforts in lifelog retrieval, and their standard methodologies are reviewed in Section 2.1. After that, Section 2.2 discusses the general landscape of QA and relates it to lifelogging. Finally, Section 2.3 concludes this chapter by summarising the key points of interactive lifelog retrieval and how QA can be applied to lifelogging.

2.1 Lifelog Retrieval

Lifelogging is the process of tracking and storing an archive of the totality of an individual's life experiences, who is often referred as a lifelogger, through technology such as smartphones and wearable devices. Lifelogging is also considered a challenging Big Data application due to the amount of data (volume), the multimodal nature of lifelog data (variety), and the inaccuracy of sensor data (veracity), although there is no need for real-time processing (velocity) at the moment [64].

In order to exploit the potentials of lifelogging, it is necessary to develop effective lifelog retrieval systems that allow users to access and explore their lifelogs effortlessly. These systems serve as a gateway to transform the complex, unstructured lifelog data into a meaningful resource for the user. In this section, I will overview the multimodality of lifelogs as well as the technologies and devices used to capture and store lifelog data. After that, I will review the early lifelog retrieval systems that laid the foundation for the development of present lifelogging systems. Finally, I will discuss the benchmarking efforts

in lifelog retrieval, focusing on the common approaches used in different lifelog retrieval systems.

2.1.1 Lifelog Data

The multimodality of lifelog data is one of the key characteristics that distinguishes lifelog retrieval from other Information Retrieval (IR) tasks [61, 199]. The most common data types in lifelogging include visual, audio, location, and physiological signals. These data types can be captured using different devices such as wearable cameras, smartphones, and wearable sensors. The following sections will discuss the different data types, the devices used to capture them, and their potential applications in lifelogging.

Visual

Wearable cameras are the most common devices used for lifelogging due to their ability to capture images and videos from the user's perspective. They are typically worn around the neck or clipped to the user's clothing. SenseCam, OMG Autographer, Narrative Clip, Google Glass, and GoPro are among the most popular wearable cameras used for lifelogging. Smartphones, with their high-quality cameras and other sensors, have also been investigated as an alternative solution to wearable cameras [65]. Photographs and videos provide a highly valuable source of information for users to recall past experiences and memories. Their visual nature also helps to communicate information to the user. As such, visual data is the most prevalent data type in lifelogging.

Audio

Audio can also be a part of lifelogging data. Devices from wearable audio recorders, smartphones, or video cameras with a microphone can all be utilised for this purpose. For example, in the MyLifeBits project [59], the author used a wearable microphone to capture audio data.

Location

Location data is captured using GPS-enabled devices such as smartphones and wearable cameras. They can be categorised into two types: GPS coordinates (latitude and longitude) and semantic locations (for example, Dublin City University, home, work, etc.). Location data can be used to infer the user's activities and the context of the lifelog data [124]. For example, the semantic location of 'coffee shop' might suggest that the user is buying a coffee, and the high speed of the user's movement suggests the user is in a vehicle. Interests, lifestyles, and preferences can also be inferred from location data [101].

Physiological Signals

Physiological signals collected by wearable sensors, such as the Empatica E4 wristband, provide a more in-depth understanding of an individual's well-being and emotional responses during daily activities. Lifelog data can provide insights into the user's emotional state and level of stress, or periods of increased physical activity by monitoring parameters such as heart rate variability, skin temperature, blood volume pulse, and galvanic skin response [12, 191]. Such data can be utilised to examine not only the immediate context of an individual's experiences but also long-term health patterns and behavioural trends.

Other Modalities

Lifelogging data may also include the user's social media activities, emails, keyboard and mouse activities, and calendar events [234].

These data can be used to infer the user's social interactions and the user's daily activities. However, these data are not considered in this thesis due to the lack of publicly available datasets.

2.1.2 Early Lifelog Retrieval Systems

In this section, I will review the early lifelog retrieval systems that laid the foundation for the development of present lifelogging systems.

SenseCam Photo Viewer

Microsoft's SenseCam is a small wearable camera that contains various sensors to detect light levels, temperature, motion, and the presence of people. In order to manage and replay the images and sensor data captured by the SenseCam, the SenseCam Photo Viewer [82] was developed and contributed much to early research in lifelogging. At the core, the system has a simple interface of an image slideshow that allows the user to play back the image sequences, as well as pause and rewind the playback. It also supports bookmarking, annotating, and deleting images from the database. The sensor data can be viewed with the images in a separate window in basic line charts, without any analysis performed on the data and no way to filter the data. Figure 2.1 shows the SenseCam Photo Viewer interface. Despite its simplicity, the SenseCam Photo Viewer has been used in many early lifelogging research to support memory rehabilitation [21, 93, 188], and its temporal browsing interaction style is still used in many lifelog retrieval systems today [99, 122, 215, 240].

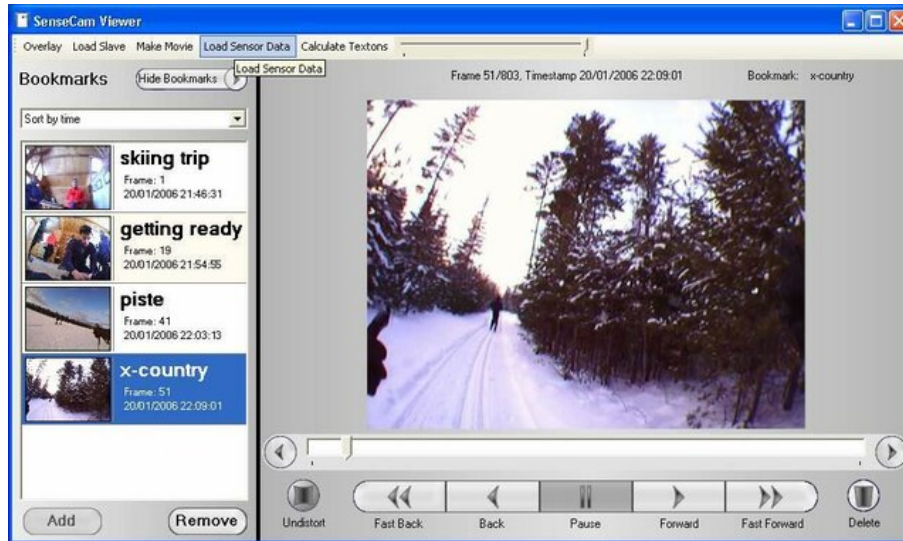


Figure 2.1: SenseCam Photo Viewer interface as reported in [81].

MyLifeBits — The first lifelog retrieval system

As mentioned previously in Chapter 1, the MyLifeBits project [59] is generally considered the first lifelog retrieval system, which was developed by Microsoft Research in 2001 as an attempt to fulfill the vision of Vannevar Bush’s Memex [24]. The project aimed to create a ‘personal database for everything’ by storing all the digital information of the user’s life, including emails, contacts, documents, events, web pages, scanned images, audio, and video recordings. The core of the system is a SQL Server database with a simple database scheme that stores the metadata of the content, for example, type, size, creation date, last modified date, and a short description; as well as the annotation and collection links. Simple filters based on the metadata (location, time) are provided to allow the user to search for the desired content. Full-text search is also supported to search for content based on the corresponding annotations. Furthermore, the user can refine, or pivot the search results, which is possible due to the links between the content. The system allows the user to view the search content in variable-sized thumbnails with multiple views such as detail, thumbnail, timeline, and clustered-time. First, the detail view shows the content in a list with their properties. Second, the thumbnail view uses a grid of thumbnails to display the content. Next, the timeline view, as seen in Figure 2.2 displays the content on a linear timescale, and the distribution of content is visualised underneath the timeline. Lastly, the clustered-time view is similar to the timeline view, but the content is clustered into groups based on their time proximity (same year, month, day).

MyLifeBits is accompanied by a set of tools to organise, access, enrich and report about the data. For instance, the annotation interface, as shown in Figure 2.2, allows the user to add annotations to the selected content which can be used for later retrieval.

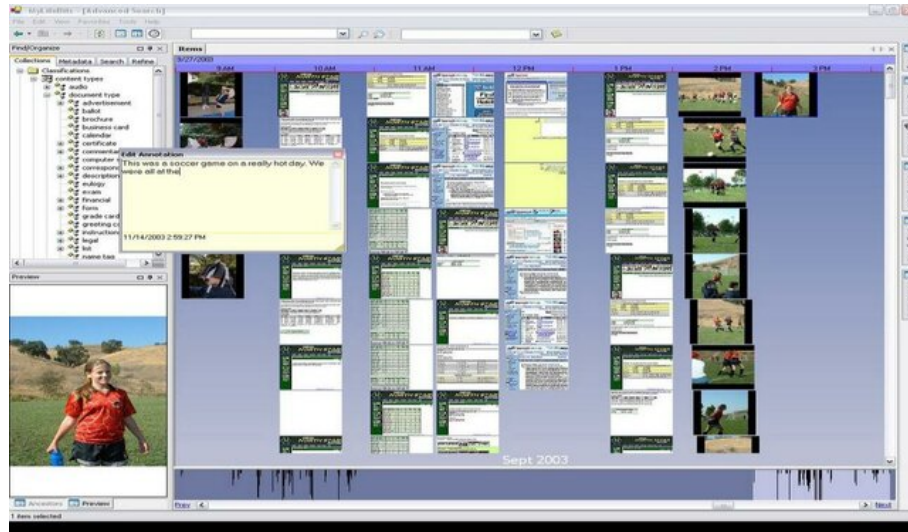


Figure 2.2: MyLifeBits query result interface.

Another tool for data visualisation is the map interface in Figure 2.3, where location data from corresponding photographs can be visualised. Map interactions such as zooming and panning will issue a new query for photos from the region of the visible map to be shown in the corresponding pane. The user can also enter a location and a time range to filter the photos, and the map will be updated accordingly. Trip detection is also supported to automatically detect and visualise the user’s trips in a slideshow. The user can also manually add or remove photos from the trip.

Overall, MyLifeBits is a comprehensive system that provides a wide range of features to support the user in managing and exploring their lifelogs. However, MyLifeBits’ complexity and multiple-step interactions could be a great barrier for users to adopt the system. Moreover, the search mechanism is limited to filters and text searches over the annotations, which requires considerable effort from the user to annotate the data.

SenseCam Visual Diary

Developed by Lee et al.[112], the SenseCam Visual Diary is a prototype lifelog browsing engine that addresses the vast amount of images that Microsoft’s SenseCam can capture. The authors applied various image analysis techniques to automatically structure and index the images, and provide a multimodal faceted search interface for the user to explore the lifelog. The SenseCam Visual Diary put forward the importance of ‘events’ in lifelog (for example working at the office, talking to a friend, buying a coffee, etc.) by automatically segmenting the lifelog into distinct events using context-based sensor analysis and content-based image analysis [41]. After that, a landmark photo for each event is selected. This can be done by selecting the middle image from the event, or by averaging the visual features across the entire set of images in the event and then selecting the image whose

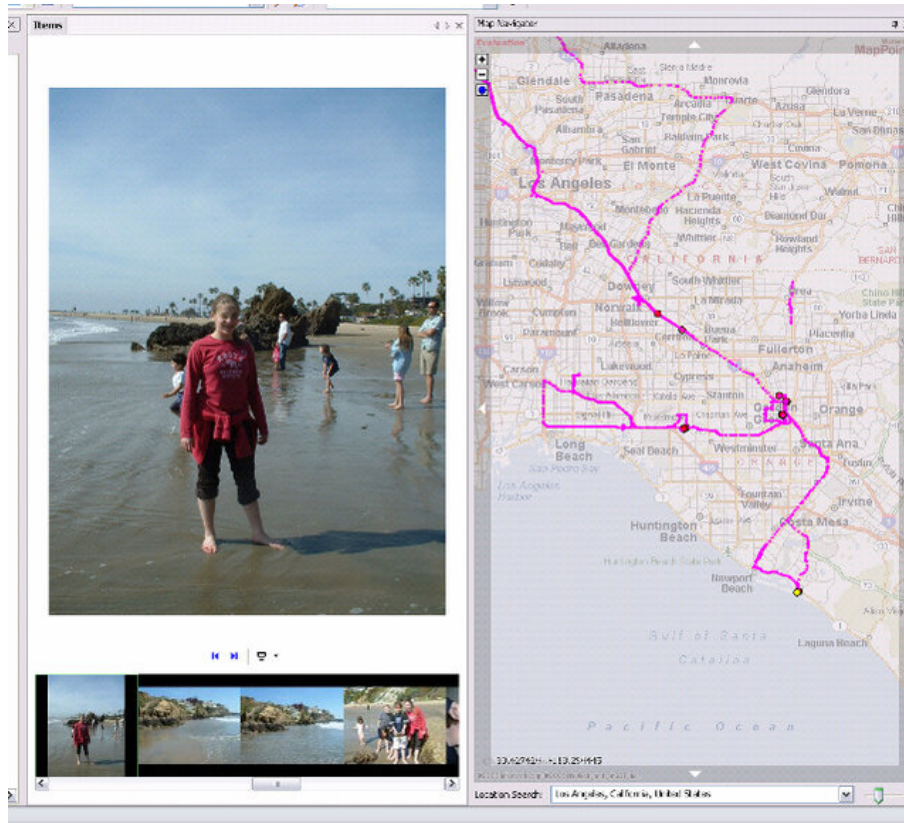


Figure 2.3: MyLifeBits' map view [13]. The map on the right shows large dots where photos are taken and small dots for GPS track points.

visual features most closely resemble the average. However, the authors also noted that there is no difference between the two approaches in practice. Finally, a novelty detection algorithm is applied to the low-level MPEG-7 visual features of the images to define the 'interestingness' of the event over a period of time, which is then shown in the user interface by varying-sized thumbnails as shown in Figure 2.4.

The user can interact with the system by choosing a date to view a breakdown of events on that day. Among the events displayed, those with higher novelty scores appear in larger thumbnails. Hovering over an event thumbnail reveals all the images in the event as a slideshow (default speed is 10 photos per second). Additionally, the event segmentation algorithm is customisable by a slider at the top of the screen, allowing the user to adjust the number of events shown. Bookmarking an event as a 'favourite' is supported so that such an event can be easily accessed later. Another interesting feature is the 'Find Similar' option, which retrieves all similar events to the selected one, which are then presented on the right column of the screen. Furthermore, annotations can be added, edited, or deleted for each event. Text-search for annotations, similar to MyLifeBits, is also available.

SenseCam Visual Diary contributed greatly to the field of lifelogging by introducing the concept of 'events' and promoting the importance of event segmentation to support

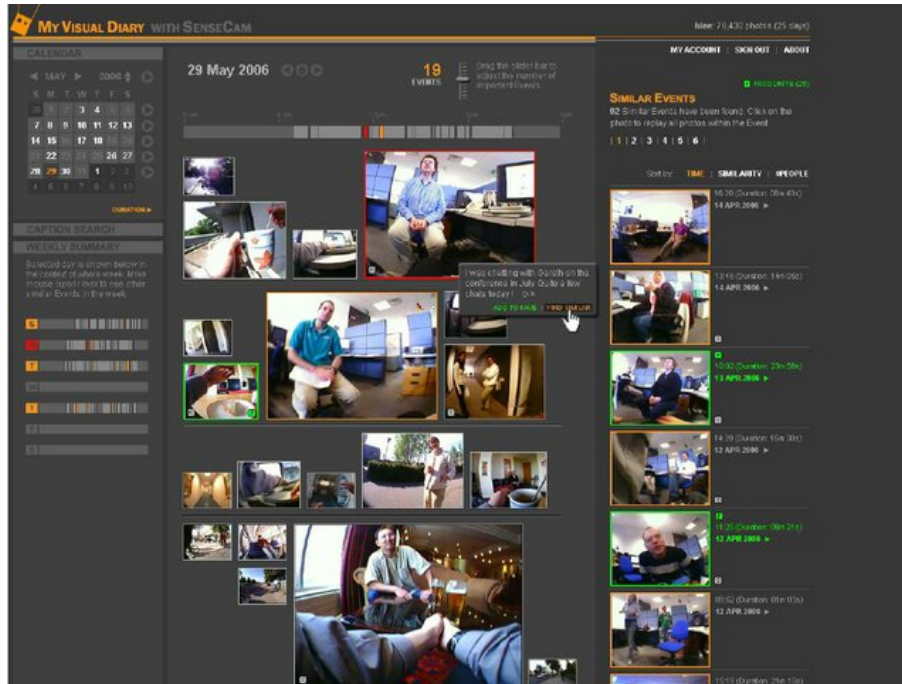


Figure 2.4: SenseCam Visual Diary interface as reported in [112].

lifelog analysis. However, similar to SenseCam Photo Viewer, it is limited to browsing rather than searching. While the system offers great support for the user to explore their lifelog through the calendar navigation and similar events scheme, this is not scalable for larger lifelog archives.

First Faceted Search Engine for Lifelog

To address the scalability issue of SenseCam Visual Diary, Doherty et al. [43] developed a faceted search system, providing users with a more efficient way to explore their lifelog data. This system, built on top of the SenseCam Visual Diary, offers various search axes: *where* (location, altitude temperature); *when* (calendar selection, prev/next day browsing, season, year, day/night, time of day, and month); *what* (visual appearance, bright/dark, important/routine, semantic concepts [42] like eating, working on PC, etc.); and *who* (estimated number of people in scene based on face detection). Users can easily select one or more facets to filter their events, making the search process more efficient.

The experiment's results suggested the effectiveness of images, rather than the sensor data, for event search, which aligns with previous findings [93]. Strictly defined boundaries between events were deemed unnecessary, as events acted as a quick navigation towards the relevant images. Consequently, there hasn't been much emphasis on event segmentation since then. The most crucial finding was that the faceted search system outperformed the SenseCam Visual Diary significantly, taking only 127 seconds compared to 774 seconds

of browsing to find the desired content. However, the authors acknowledged that even 127 seconds was still unacceptable to users who expect prompt access to relevant lifelog information. This highlighted the increasing need for search-based systems in lifelogging, which is the direction the field has been heading towards since then.

Being the first system that supports faceted search for lifelog data, this system is a great example of how faceted search can be applied to lifelogging. As a result, faceted search has been widely adopted in many lifelog retrieval systems since then [240].

2.1.3 Benchmarking Challenges

Benchmarking is an important part of research, as it allows researchers to compare their work with others and evaluate the effectiveness of their system. In the context of lifelogging, having a standardised benchmarking activity becomes even more vital as it is a relatively new field with mostly different small-scale experiments, often years apart, and operating on vastly different datasets. It was not until 2016 that the first lifelog benchmarking challenge took place, where international participants developed systems to address the pilot task in NTCIR-12 [63]. This marked a significant step in establishing a robust framework for evaluating lifelogging methodologies and motivating further advancements in the field. Since then, different lifelog challenges have been established with distinctive evaluation metrics to assess lifelog systems. In this section, I will discuss the three most important lifelogging benchmarking challenges to date: NTCIR, ImageCLEF, and the Lifelog Search Challenge (LSC).

NTCIR LAST — Lifelog Semantic Access Task

NTCIR (NII Testbeds and Community for Information access Research) is an evaluation forum that aims to advance research in information access technologies, such as IR, question answering, and summarisation. NTCIR has been running since 1999, with the first lifelogging task introduced NTCIR-12 [63] in 2016. Lifelog retrieval has been a subtask of NTCIR Lifelog challenge since then under the name Lifelog Semantic Access Task (LAST).

The NTCIR LAST is a known-item search (KIS) task, where participants are required to retrieve a number of specific events or activities in a lifelogger’s life. The task can be done in an interactive or automatic manner. The test collection consists of a large volume of lifelog data, including images taken from wearable cameras, an XML description of the semantic locations, for example home, work, and airport; and the lifelogger’s activities, e.g walking, running, and transport; and additional visual concepts detected from a CAFFE CNN-based object detector [89]. In later years, more data were included such as GPS location and biometrics data from sensors (for example, heart-rate monitors). In this task, the evaluation metrics used are from TREC, which contains a set of metrics that are commonly used in IR. The two most important metrics are Mean Average Precision

(MAP) and Normalised Discounted Cumulative Gain (NDCG).

ImageCLEF LMRT — Lifelog Moment Retrieval Task

ImageCLEF is a part of the Conference and Labs of the Evaluation Forum (CLEF), which is an organisation that promotes research, innovation, and development of information access systems. Lifelog Moment Retrieval Task (LMRT) in ImageCLEF challenges [33–35, 155] was established for lifelogging to gain more attraction from the research community. This challenge is similar to NTCIR LAST, which also focuses on known-item search. However, different evaluation metrics are used, namely:

- Cluster Recall at X (CR@X) — a metric that assesses how many different clusters from the ground truth are represented among the top X results;
- Precision at X (P@X) — measures the number of relevant photos among the top X results;
- F1-measure at X (F1@X) — the harmonic mean of the previous two

Various cut-off points X are considered, such as X=5, 10, 20, 30, 40, and 50, where F1@10 is the ranking metric.

ACM LSC — Lifelog Search Challenge

The Lifelog Search Challenge (LSC) [68] is a part of the ACM International Conference on Multimedia Retrieval (ICMR). The challenge was first introduced in 2018 and has attracted the largest number of participants among all lifelogging challenges. The focus of the LSC is to evaluate the performance of interactive lifelog search engines *in real time*. Different systems compete with each other in a live/virtual environment, where the participants are given a set of queries and limited time to find the relevant images.

The LSC started with one type of task, which is the **known-item search (KIS)** task. For each KIS task, a time limit of five minutes is given to the participants to find the relevant images. The organiser will gradually provide an additional hint every 30 seconds (at 0, 30, 60, 90, 120, and 150 seconds) to imitate the real-life scenarios that people usually remember slowly over time. An example of LSC’s KIS task can be seen in Table 2.1 some images from the ground truth are shown in Figure 2.5.

Each participating team has to find *any* relevant image for each task and submit it to a host server [180]. The host server keeps track of time with a countdown clock and then assesses the submissions against the ground truth to evaluate their accuracy.

In later years, more types of tasks were introduced, namely **Ad-hoc** and **Question Answering** tasks. In LSC’22, the first Ad-hoc task was introduced, in which participants were tasked with submitting as many images as possible that were relevant to the given

Table 2.1: An example KIS task from LSC’20 [212]. Task 1 with its temporally advancing descriptors, which were revealed at 30-second intervals. After 150 seconds, the full description is shown for another 150 seconds until the end of the task.

Time	Text
0s	I was building a computer alone in the early morning on a Friday. . .
30s	I was building a computer alone in the early morning on a Friday at a desk. . .
60s	I was building a computer alone in the early morning on a Friday at a desk with a blue background. . .
90s	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. . .
120s	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. . .
150s	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.



Numbers of images: 92
Time: 07:24AM - 09:27AM
Date: 13/03/2015 (Friday)
Semantic Name: Dublin City University (DCU)

Figure 2.5: Some lifelog images as ground truth from Task I in Table 2.1

query within the time limit. Because no ground truth was provided in advance, human judges were required to assess the relevance of the submissions. The Question Answering task was introduced in LSC’22 and the same submission/evaluation mechanism as the KIS task was used, in which the answer was in the form of an image. It was not until 2023 that the task was fully realised, in which the answer was in the form of a free-form text. Participants had to submit a text answer to the given question, and human judges had to determine whether or not the answer was correct. The Question Answering task has a time limit of three minutes.

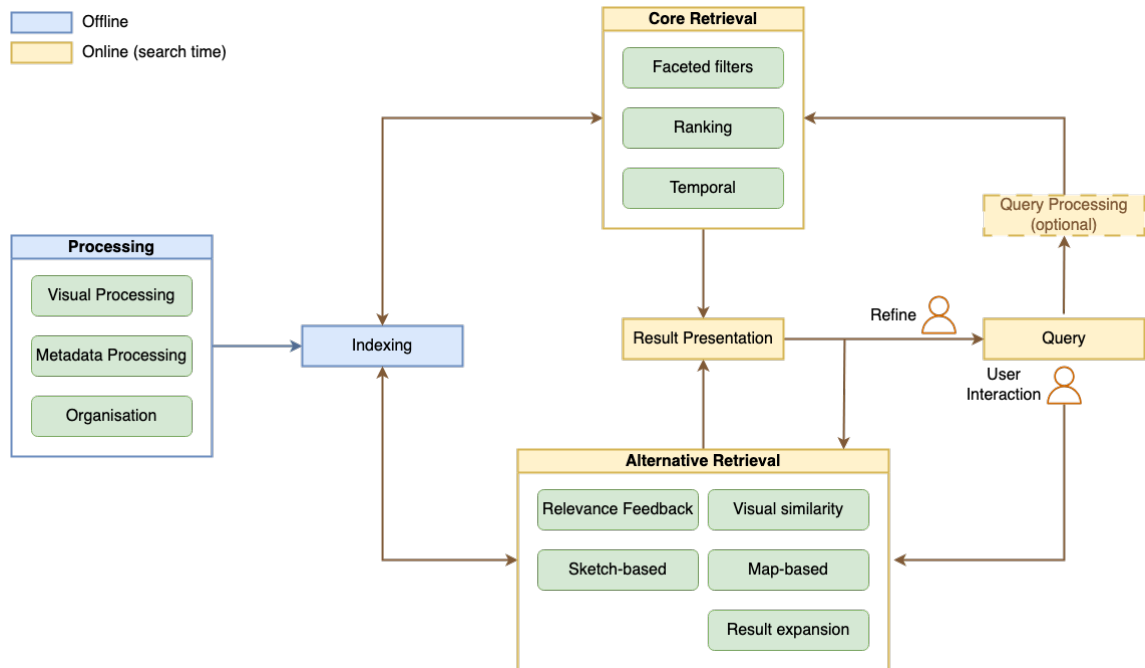


Figure 2.6: The general pipeline of a lifelog retrieval system.

2.1.4 Approaches

I now discuss the approaches taken by the systems that participated in the three challenges discussed above. Since lifelog systems are complicated and involve many different aspects, this section will be discussed based on the general pipeline as seen in Figure 2.6. Most lifelog systems following this pipeline, and the approaches taken by the systems can be categorised into the following components: processing and indexing, retrieval, and user interaction. I will now overview each component of the lifelog pipeline.

Processing and Indexing is the first step in the lifelog pipeline, where the lifelog data is processed and indexed to make it searchable. Two main types of processing will be discussed: visual processing and metadata processing. Visual processing, as in Section 2.1.4, refers to the extraction of information from visual data, which is the most dominant modality in lifelogging. Heavy use of computer vision algorithms is made to extract semantic information from visual data. In contrast, Section 2.1.4 discusses the processing of metadata, which is often overlooked by the systems. The metadata is often noisy and requires processing to extract useful information from it. However, the potential to gain insights from the metadata is enormous. After processing, the data is organised (Section 2.1.4) and indexed (Section 2.1.4) to make it searchable.

Retrieval is the second step in the lifelog pipeline, where the user's query is matched against the indexed data to retrieve the relevant lifelog data. Section 2.1.4 discusses the different approaches taken by the systems to retrieve and rank the lifelog data. Alternative methods are often used to improve the retrieval performance, such as relevance feedback

and visual similarity measures. These methods are discussed in Section 2.1.4.

Finally, User Interaction is the last step in the lifelog pipeline, where the user interacts with the system to explore and retrieve their lifelog data. Section 2.1.4 discusses the different methods used by the systems to present the results to the user. Some novel platforms and interactions are also discussed in Section 2.1.4 as they provide interesting insights into the future of lifelog retrieval systems.

Visual Processing

Due to the multimodal nature of lifelog data, the processing of lifelog data is a complex task that involves the extraction of information from different modalities. Most of the systems follow a basic processing for location, time, and sensor data. The main difference between the systems is the processing of visual data, which is arguably the most dominant modality in lifelogging and the most challenging to process. State-of-the-art techniques from the field of image processing and computer vision are exploited to extract semantic information from visual data. The extracted features can be broadly categorised into low-level, content-based, captioning, and cross-embedding features.

Low-level processing This refers to the use of *low-level features* such as colour, texture, and shape to extract information from images. These are classic features that have been used in the field of image processing for decades and are still used in lifelog systems. For example, LEMoRe [159] exploited auto colour correlogram, edge histogram, joint composite descriptor, and histogram of oriented gradients (HOG), and BiDAL [36] used colour histogram with other content-based features, to calculate visual similarity between lifelog images. Similarly, VIRET [133] utilised a colour-based distance between adjacent images to reduce redundancy in the dataset. Another use of low-level features is adopting HOG [226] to estimate the number of people in an image. It is also common to use these features to detect blurriness or covered images in an attempt to reduce the volume of data, as well as increase the quality of the retrieved images. For instance, the ZJUTCVR system [241] used a Laplacian filter to detect blur and remove covered images, defined by images with the main subject's size over 90% of the image by the authors. Another example is FuM [52] which incorporated lens calibration, followed by blurriness and color diversity detection.

Removal of low-quality images Another common recent practice in lifelog systems is an application of low-level features. It is believed that these images are less likely to be relevant to the user's query. On that note, Blind Image Quality Assessment (BIQA) was suggested by MEMORIA [174], a new system that has been participated since LSC'22, to predict the quality of images. The authors employed a deep learning Koncept512 model [84] to predict the Mean Opinion Score (MOS) of each image, which was then used to filter out low-quality images with a threshold.

Concept-based features Due to the advancements of computer vision algorithms,

we can extract *content-based features* from images using various deep convolutional neural networks (CNNs). These features are more semantic than low-level features and can be used to describe the content of an image. Some examples of CNNs that are commonly used in lifelog systems are AlexNet [105], VGG [196], GoogLeNet [200], and ResNet [73], trained on various datasets such as ImageNet [37], OpenImages [106], Places365 [238], and Visual Genome [104]. These models serve various purposes in lifelog systems, acting as image classifiers [40, 125, 182], object detectors [40, 110, 111, 125, 156, 159, 176], object recognition models [226], and scene recognition models [40, 156, 226]. One way of using computer vision models is to extract the features from the last layer of the model and use them as a representation of the image, which can be used directly as a visual similarity measure [36, 147] or as a feature to train a classifier for relevance feedback [95, 241]. However, the most popular approach is to use these models to obtain a set of semantic lifelog tags, or ‘concepts’ (e.g ‘person’, ‘car’, ‘building’, etc.) from the images, which are then indexed and used for retrieval. Additionally, other sources of lifelog concepts are also available from services such as Microsoft Vision API¹ and Google Cloud Vision². Optical character recognition (OCR) has also been used to extract text from images, which can be seen as another source of concepts. Text recognition started to gain popularity in lifelog systems since LSC’20 where LifeSeeker [110] and FIRST [215] employed various models to obtain brand names, product names in a shop, and street names from lifelog images. In LSC’21, OCR proved to be very useful, with half of the queries could be solved by OCR alone [212]. Overall, this approach of extracting concepts from images, referred to as **concept-based retrieval**, is widely adopted in many lifelog systems due to its ease of implementation and the availability of pretrained models. In fact, nearly all the systems participating in the challenges discussed in this chapter employed this approach.

Captioning Another approach is to use the advancement in the field of IR and turn images into text documents by **generating captions** for them [75, 176, 214]. However, captioning does not seem to be a popular approach in lifelog systems, with only a few systems using it. This is likely due to the fact that captioning is a very challenging task, and the generated captions are not always accurate, especially for lifelog images.

Cross-embedding models In LSC’20, BIDL [142] put forward a novel approach to lifelog retrieval by transferring the text query into a visual embedding space. Specifically, a seq2seq model, initialised with a pretrained BERT [38] was used to extract concepts from the text query, whose embeddings were then transferred to the visual embedding space using an attention-based algorithm. The embeddings were then used to retrieve the relevant images using cosine similarity. This approach is referred to as *embedding-based retrieval*. Similarly, in the same year, FIRST [215] proposed an encoder-decoder architecture to embed text and images into a common space, with a reconstruction loss to

¹<https://azure.microsoft.com/en-us/services/cognitive-services/computer-vision/>

²<https://cloud.google.com/vision/>

retain the original information. Another example is LifeSeeker [110] using a pretrained W2VV [44] to map the text query to a feature vector similar to that of ResNet [73]. However, it was not until LSC’21 that the importance of cross-embedding models was realised due to the revolutionary results they have achieved in the field of computer vision [212]. These models are trained to embed different modalities into a common space, where the similarity between the modalities can be measured. Large-scaled pretrained models such as Contrastive Language-Image Pre-Training (CLIP)[167] and Bootstrapping Language-Image Pre-training (BLIP)[123] have been utilised since LSC’21 [212] to extract the visual features from the lifelog images and textual features from search queries, then rank the images based on cosine similarity. This **embedding-based retrieval** approach allows a more user-friendly experience by allowing users to search for images using natural language queries and significantly improves the retrieval performance [4, 207].

Metadata Processing

Very little processing for metadata was done by the many systems participating in the challenges because the organisers normally provide the metadata in a standardised CSV format. However, the provided data is still very noisy (considering the *veracity* aspect of lifelogs). For example, missing GPS coordinates is a common issue in lifelogging datasets, which can be handled by the systems in different ways. Some systems ignored the missing GPS coordinates and used the available data only. This had little effect on the retrieval performance because most queries focus on visual hints. However, I believe processing metadata is still important because the potential to gain insights from the metadata is enormous.

Time and Location The two most commonly processed metadata are time and location. Regarding time information, LIG-MRM [182] used a simple rule-based process to annotate the images with time-of-day labels (morning, noon, evening). As for location information, clustering is a common approach to infer the semantic location of an image. For instance, LEMoRe [159] manually clustered locations and replaced the location name with a general location name (*Aldi* or *Tesco* would be changed to *supermarket*) in order to simplify the lifelog. PGB [226] employed a stay point detection algorithm (using D-Star clustering) and an important location detection algorithm (using DBSCAN). To fill in the missing data, THUIR [122] used a simple rule-based approach to forward fill the missing data if the lifelogger is not moving; LifeGraph [177] employed temporal interpolation; and Myscéal [206] (my work) used both interpolation and clustering to infer the missing semantic locations. Some systems incorporated both location and time information to infer new data, such as LifeSeeker [156], which derived region and country information from the timezone. In its later version, LifeSeeker [110] also manually labelled the images with ‘areas’ within a location (for example kitchen, bedroom, office, etc.) to create a more fine-grained location hierarchy (of three levels: country, location, and area). After

that, the remaining unlabelled images are labelled using a simple rule-based approach considering the visual similarity and scene concepts. Semantic locations can also be used to add more concepts to the image, such as adding ‘coffee shop’ to an image taken at ‘Costa Coffee’[96]. Recently, MyEachtra [213] (my work) linked visual features with GPS coordinates to infer the semantic locations in an automatic manner.

Biometrics Only a few of them processed biometrics sensor data, often by binning [214] or categorising (e.g into resting, normal, and physically active)[8, 96]. Music listening data was also processed by LifeSeeker [151] to infer the user’s mood based on the arousal and valence detection algorithm using Spotify’s API³.

Organisation

Due to the temporal nature of lifelog data, it is straightforward to organise the data by chronological order as proposed by Zhou’s baseline system [239] for NTCIR-12. In that system, the minute was used as the basic unit. A few systems used a more sophisticated approach to organise the data. For example, [95] split the images into the subdirectories using location and activity. Segmentation is also a common approach to organise the data. Most systems that included segmentation used a simple approach by comparing the visual similarity between two consecutive images [96, 205]. MyEachtra [213] clustered the GPS coordinates to infer the semantic locations and then organise the data by the semantic locations.

LifeGraph [177] offered a novel approach to organise the data by using a graph structure, with images as the centre point of the schema. The graph was constructed by linking the images with the metadata and detected concepts, then extended with Wikidata [219] and COEL (Classification of Everyday Living)[23]. However, the authors later acknowledged that COEL played an insignificant role in the query expansion and that Wikidata was a more useful source of concepts [178].

Indexing

Some database services were commonly used to index the lifelog data. These services provide a simple way to index and search the data in optimised settings. For example, LEMoRe employed Lucene Image Retrieval engine (LIRE)[140], Myscéal [205] and LifeSeeker [110] exploited Elasticsearch⁴, and Vitriivr [179] used CottontailDB [58] as the storage backend. Accommodating the graph-based approach in LifeGraph [177], the authors initially employed BlazeGraph⁵ to store the graph data and used SPARQL [165] to query the data. However, CottontailDB was later used to replace BlazeGraph due to its

³<https://developer.spotify.com/documentation/web-api>

⁴<https://www.elastic.co/>

⁵<https://www.blazegraph.com/>

better performance [178]. On the other hand, some systems used their own indexing methods, for example, hash tables [111]. FAISS [90] is also a popular library for indexing and searching vectors in high-dimensional spaces, which was used by many embedding-based systems [5, 142, 187].

Ranking and Retrieval

The most distinguishing factor between different lifelog retrieval systems is their retrieval mechanism, which is the core of the system and affects the processing, indexing, and presentation of the results.

Filters These are used notoriously as they have been proven to be effective for searching in the early, multi-faceted lifelog system [43] mentioned previously. In the first NTCIR challenge [63], a baseline system developed by Zhou et al.[239] also showed the efficiency and ease of implementation of faceted filters. They can be applied to various modalities with some examples including:

- time of day: morning, noon, evening, night
- day of the week: Monday, Tuesday, etc.
- month: January, February, March, etc.
- year: 2015, 2018, 2021
- location: home, work, school, Aldi, Tesco, etc.
- number of people
- biometrics: binned values of heart rate, steps, etc.
- lifelog concepts: scene categories (restaurant, library, etc.), activities (eating, reading, etc.), objects (food, book, etc.), which are often extracted from the image using deep learning models

Choosing filter values Most of the filter values are fixed phrases, which can be chosen from a drop-down list, a checkbox, or a slider. An example of this could be seen in lifeXplore [147]’s interface in Figure 2.7. However, some systems allowed users to enter free text and use autocomplete to suggest the available values for the lifelog concepts [239]. This is necessary as the number of concepts from the content-based models is increasingly large, which makes it difficult to present them in a drop-down list. Other than autocomplete, concept suggestion [27, 133, 151, 159, 206] or query expansion [111, 149, 214], using various sources like WordNet [160], ConceptNet [198], and Thesaurus.com, were also used to help users formulate queries. LSTM was adapted predict to relevant concepts in [1]. Similarly, VieLens [149] and BIDAL [142] employed BERT [38] to find the most similar concepts to



Figure 2.7: Faceted filters in lifeXplore [147]

the query terms. In addition, Myscéal used a free-text query form, which allowed users to enter any text and the system would extract the filter values from the query and perform query expansion to improve the retrieval performance. Following Myscéal, LifeSeeker also added a text parser in its later version [151] to process the query using NLP techniques.

Scoring and ranking using concepts Filtering is helpful in removing the irrelevant data from the result set; however, as with any retrieval system, ranking the results by some relevance scores is the most important step. Some filters can be used to score the data based on the number of filters matched [239], on a function that extends TF-IDF [8, 27, 52, 122, 205, 226], and on the confidence scores of the deep learning model [40]. TF-IDF is a common technique in IR to score the relevance of a document to a query. It is calculated based on the term frequency (TF) and inverse document frequency (IDF) of the query terms. TF scores are not as useful as they are in the field of IR, since the terms (concepts) are oftentimes not repeated in a document (image). Therefore, the TF scores are often replaced by the confidence scores of the concepts extracted from various computer vision models [40]. The area of the object (or its bounding box) can also be exploited as in aTFIDF, proposed by Myscéal [205]. Alternatively, vitivr [75] aimed to highlight the different importance of the query terms by proposing a staged querying mechanism, where the search was performed in steps in which the next query was a subset of the previous one.

Another way of scoring the images based on their concepts is creating a bag-of-words (BoW) representation of the images and using the cosine similarity between the query and the images [110, 156, 240]. This BoW feature can also be used for measuring visual similarity [214], similar to low-level and content-based features mentioned in the processing

step.

Cross-embedding similarity As previously discussed, crossmodal embedding models have offered a new way of measuring the similarity between images and text in lifelog retrieval systems. Cosine similarity between the search query and the images is directly used to rank the result. Some optimisation methods such as KNN search [207] or FAISS [90] can be used to speed up the search as in the case of lifeXplore [187] and Memento [5]. Although there is some effort of finetuning the embedding models on lifelog model [211], large-scale pretrained models are more robust and are often used, either directly or in a weighted ensemble [5].

Temporal search Some groups also supported temporal search, which allows users to combine multiple queries that are temporally related. For example, the user can search for ‘eating apple *before* watching TV’. SOMHunter [132] supported two temporally ordered queries, where a fix-size neighborhood of the first query’s result is considered for the second query. Myscéal [205] and Memento [4] handled up to three ordered queries, where the supplementary queries (before and after) are performed conditionally on the main one and the results are re-ranked. An arbitrary number of queries could be searched in Vitivr [199] which fused scores of multiple temporally ordered queries in a late fusion step. The relevance feedback mechanism, which will be discussed in the next section, in Exquisitor [97] also allowed temporal search where multiple classifiers were merged using union, intersection, and difference operations, or based on temporal constraints.

Alternative Retrieval Methods

Relevance feedback It is notable that some systems employed a *relevance feedback mechanism* for retrieval [95, 98, 102, 145, 241]. This is a common technique in IR, which allows users to provide feedback to the system on search results. Based on the result list, the user can label the images as relevant (+) or irrelevant (-), and then prompt the system to train a new or existing classifier to retrieve a new set of images. The process repeats until the user is satisfied with the result or finds the needed images. Exquisitor [99] is a good example of this approach, which can be seen in Figure 2.8. Oftentimes, a query initialisation using filters or text query was used to start the search, which was then refined by relevance feedback. In SOMHunter [145], the user only needed to choose the relevant images and the irrelevant ones would be randomly sampled from the unselected images. In another system [27], the result could be refined by performing a nearest neighbour search on any selected (irrelevant) images to find similar images and removing them from the result list. Exclusion of concepts is also a common idea, where the user can specify the concepts that they do not want to see in the result list [178, 208].

Visual similarity As seen in SenseCam Visual Diary, *visual similarity* has been a popular method to use in conjunction with other retrieval methods. It can be used to arrange the result list [118, 145], or more often, to provide an alternative way for users

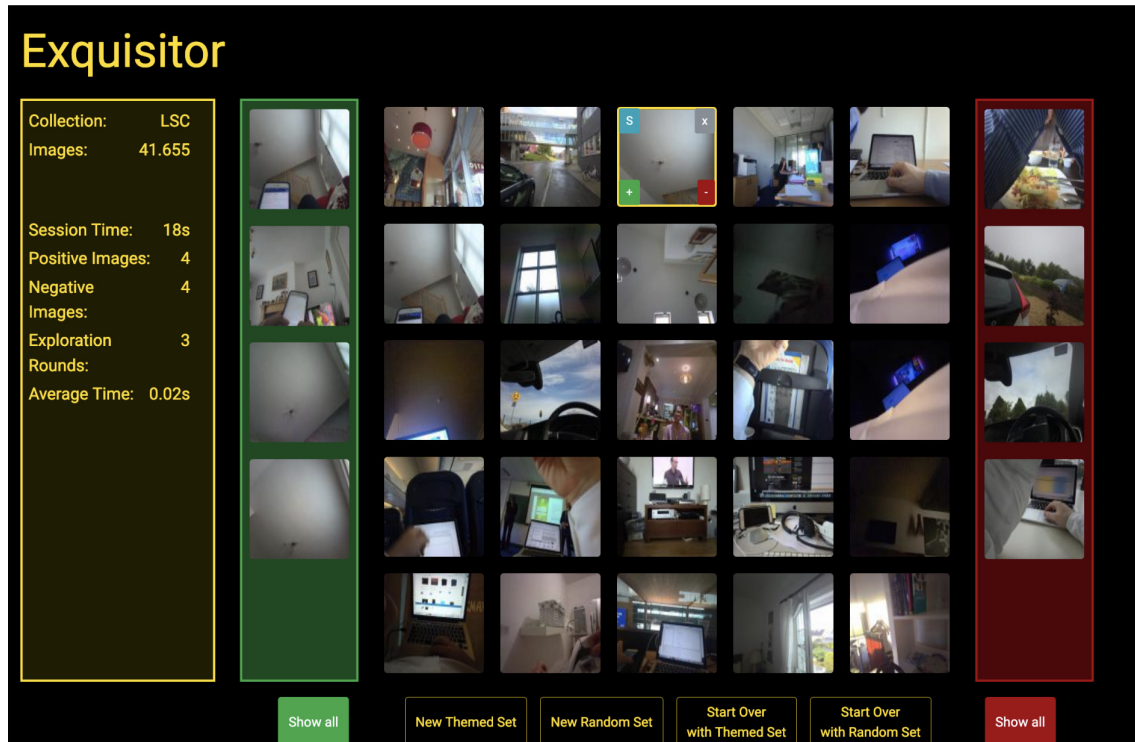


Figure 2.8: Relevance feedback in Exquisitor [99]

to explore the lifelog archive. Visual similarity can be calculated based on low-level features such as colour histogram [36, 96, 159] or SIFT [96, 147, 205, 240], content-based features [36], or cross-modal embedding [207].

Sketch-based search *Sketch-based search* is another interesting method that is seen in some systems that originate from the video search community such as lifeExplore [147], VIRET [133], and vitivr [176]. It allows users to sketch the image they are looking for, and the system will return the images that are visually similar to the sketch (see Figure 2.9). This method is useful when the user cannot describe the image in words, however, it is not as popular since the user is unlikely to know what the target image looks like.

Map-based search *Map-based search* is popular in many systems such as lifeExplore [119], Myscéal [205], and vitivr [75], where the user can draw a rectangle on the map to narrow the search space down to only the moments that happened inside that area. A common choice of libraries for map-based search is Leaflet⁶.

Result expansion To tackle ad-hoc retrieval, some systems expanded the result list by clustering the lifelog images offline based on their visual similarity. For example, during the retrieval, given a user-selected image, the result list can be extended by adding the images that belong to the same cluster of the selected image. This approach is referred to as Watershed in BIDADL [142].

⁶<https://leafletjs.com/>

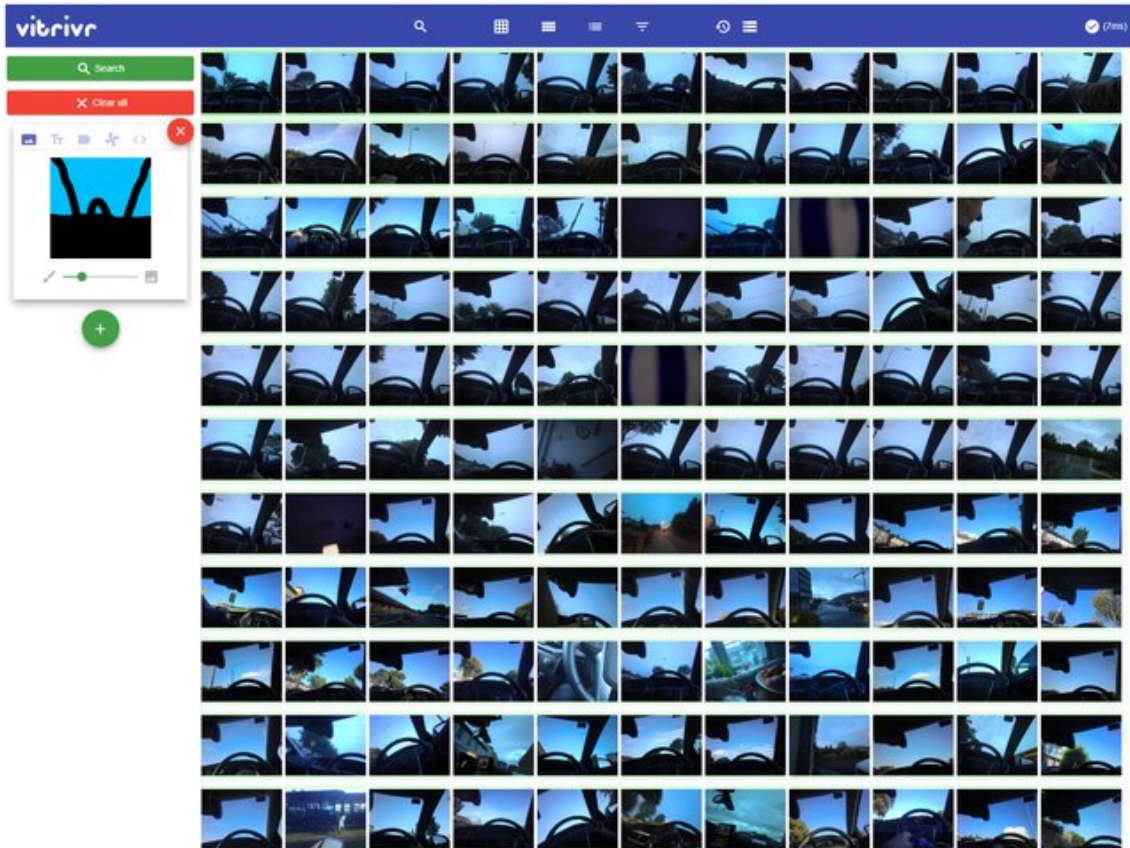


Figure 2.9: Sketch-based search in vibriv [75]

Result Presentation

Ordering the result list The conventional way of presenting the result list is in a grid view, where the images are arranged in a grid with a fixed number of columns, with the relevance scores decreasing from left to right and from top to bottom. However, some systems have explored other ways of presenting the result list. For example, lifeExplore [147] and SOMHunter [145] explored Self-Organising Maps (SOM) to arrange the images in a 2D map, where the images are clustered based on their visual similarity. In LSC'20, lifeExplore [119] proposed an autopilot navigation mode to guide the focus of the user through the result list.

Clustering and grouping In order to reduce the visual cluster due to the number of similar images in lifelog data, some extent of event clustering can be adapted to group similar images that belong to the same event. For example, Myscéal [205] performed offline event segmentation and only showed the highest-ranked image from each event in the result list. On the other hand, LifeSeeker favoured a dynamic approach by clustering the images based on their visual similarity and temporal proximity during the retrieval [151] and showed the top-3 images from each cluster in the result list. Similarly, to reduce the

vertigo effect in a VR environment, VRLE [47] proposed a horizontal grid of events, where each event contains up to nine images, chosen by the system based on their relevance scores (the number of concepts that are shared between the image and the search query in this case), and arranged temporally.

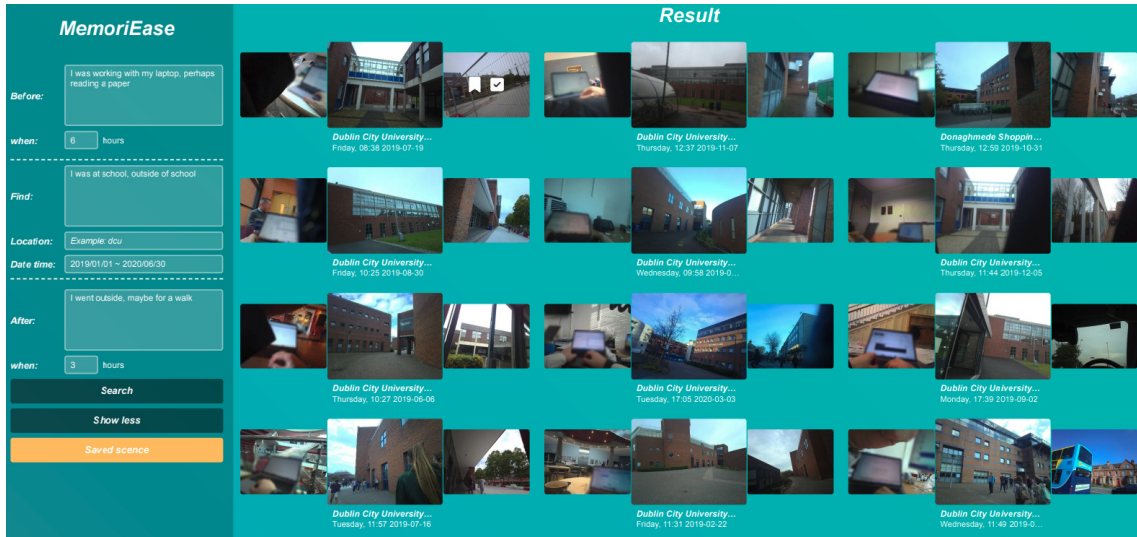


Figure 2.10: Results in triplets in MemoriEase [216].

Temporal highlighting Addressing the temporal property of lifelogs, Myscéal and MemoriEase [216] proposed showing the result in triplets, where the immediately previous and next events are shown alongside the target event, as seen in Figure 2.10. This is to provide more context to the user and allow them to explore the lifelog archive more easily. However, this approach is not always relevant to the query, especially when the query is not temporally related. Therefore, in LSC’22, E-Myscéal (second update of Myscéal) adopted a dynamic approach by showing the results in pairs or triplets only when the temporal queries are specified.

Analysis of the results Memento [4] offered visualisation of the results by distribution charts (see Figure 2.11). The charts show the distribution of the results on various dimensions such as time and location and allow the user to modify the filters directly by interacting with them. Similarly, transitional graph-based visualisation was also proposed by LifeSeeker [110] to show the location transitions between the images in the result list (for example starting from *home*, going to the *airport*, to the *supermarket*, etc.). This is useful for queries that involve transportation such as ‘flying to London’ or ‘driving to the restaurant’. The graph can be seen in Figure 2.12 and the user can interact with it by clicking on the nodes.

Timeline view Another important aspect of a lifelog retrieval system is the ability to *browse* the lifelog archive in chronological order, a.k.a. *timeline view*. This is useful when the user wants to explore the lifelog archive without a specific goal in mind or when



Figure 2.11: Data filtering visualisation in Memento [4]

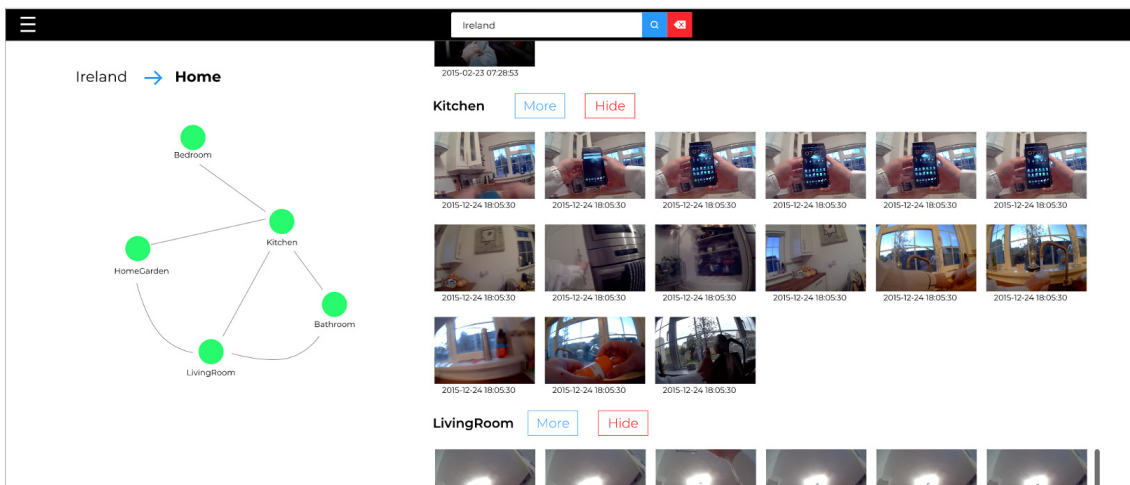


Figure 2.12: Location transitional graph with fine-grained location hierarchy in LifeSeeker [110].

they want to find a specific event that happened at a certain time. Most of the systems supported timeline view [99, 122, 207, 215], with different levels of granularity and designs. For example, Exquisitor in LSC'20 [99] shared their insights on the necessity of timeline view and designed their temporal context view as a video player with lifelog images as thumbnails. The user can play the video to see the images in chronological order and navigate to a specific time by selecting a thumbnail underneath the video. The ability to adjust how far the thumbnails are apart is also important and can be implemented with a scaling factor as in LifeSeeker [110], a step slider as in FIRST [215], or by different hierarchical levels as in Myscéal [206].

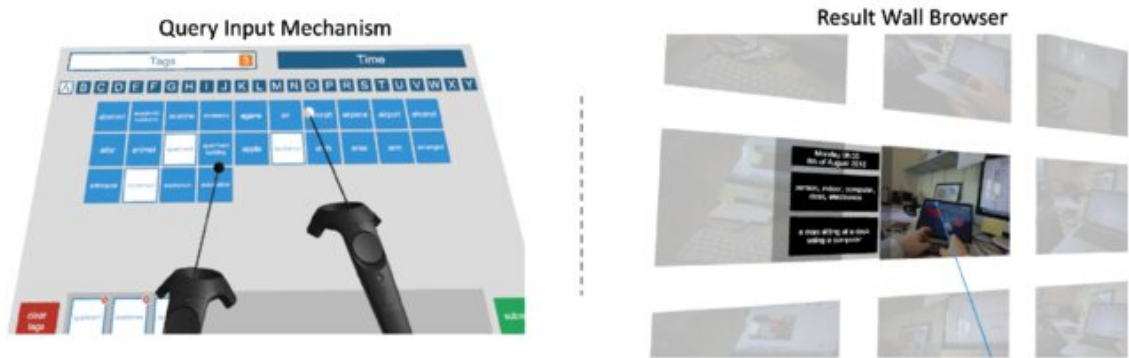


Figure 2.13: Virtual Reality Interface of UU-DCU team in LSC'18.[46].

Novel Platforms and Interactions

Most of the systems that participated in the challenges are designed for desktop computers, where the user can interact with the system using a mouse and a keyboard. However, some systems have explored other platforms and interactions. For example, the winning system at the first LSC [46] was designed for virtual reality (VR) headsets. The user can choose concept tags from a board in front of them, as Figure 2.13 shows, and the system will retrieve the images that are relevant to the selected concepts. Since then, other VR systems have been developed such as VRLE [47], vitrivr-VR [199], PhotoCube [193], ViRMA [48]. Another example of different platforms is XQC [102], which was designed for mobile interfaces, especially Android-based devices. Speech commands were also explored in some systems such as Voxento [7] and vitrivr-VR [199].

2.1.5 Discussion

This section provided an overview of the lifelog retrieval systems that have participated in the lifelog challenges since 2016. These systems have evolved significantly over the years, from simple systems that only support basic search to more complex systems that support a wide range of retrieval mechanisms and user interactions. Table 2.2 summarises the approaches used by the systems in the challenges. For the sake of brevity, I only include the systems within the top 3 in the most recent LSCs, since they are the most relevant to this dissertation.

From the above table, we can see that the most popular approach is to extract concepts from the images and index them for retrieval. This approach is simple and effective, and it will likely be used in the future. However, there are still some limitations to this approach. Firstly, the concepts are not always accurate, especially when the images are not clear or the concepts are not well-trained. Secondly, the preprocessing part can be costly and time-consuming to employ a diversity of computer vision models. Moreover, the

Table 2.2: Selected approaches used by participating systems, adapted from [212]. For each system, a reference to the paper describing the method is given. The order of the systems is based on their ranking each year. The systems in bold are the ones that I have worked on.

	winning year	concepts search	embedding	OCR	temporal query	relevance feedback	visual similarity	location visualisation	novel interaction
Myscéal [205]	LSC'20	✓			✓		✓	✓	
SomHunter [145]	LSC'20	✓		✓	✓	✓			
vitriivr [75]	LSC'20	✓			✓	✓	✓	✓	
Myscéal [206]	LSC'21	✓		✓	✓		✓	✓	
SomHunter+[132]	LSC'21	✓		✓	✓	✓			
LifeSeeker [150]	LSC'21	✓		✓	✓		✓		✓
E-Myscéal [207]	LSC'22	✓	✓	✓	✓		✓	✓	
LifeSeeker [151]	LSC'22	✓	✓	✓	✓		✓		✓
Memento [4]	LSC'22		✓	✓	✓				
lifeXplore [187]	LSC'23	✓	✓	✓			✓		✓
MyEachtra	LSC'23		✓	✓	✓		✓	✓	
Memento [5]	LSC'23		✓	✓	✓				

performance of the model relies heavily on query expansion, which is not always efficient and accurate. Cross-embedding models have shown promising results in bridging the semantic gap between the text queries and the visual content of lifelog. It also eliminates the need for searchers to be familiar with the indexed concepts.

Optical Character Recognition (OCR) is another important feature that is useful for searching in lifelog data. However, it is prone to errors, especially when the text is not clear, obscured by other objects, or arranged unconventionally such as in designed logos or posters. Therefore, it is important to have a mechanism to search for partial matches. The rise of the embedding models has somewhat reduced the need for a dedicated OCR model, since the text information is already embedded in the image embedding. However, for optimal performance, dedicated OCR models remain a favourable option.

Considering the temporal nature of lifelog data and how often the queries are often formulated based on time, temporal search capabilities play a significant role, as discussed in the previous section. Alternative search methods such as visual similarity and relevance feedback are also popular among the teams and will likely be used in the future. Another important aspect shown in the table is location visualisation. Half of the top-3 teams in the last four years have incorporated some form of location visualisation, which shows the importance of location in lifelog retrieval. Nevertheless, it is not always a strict requirement and can be supplemented by other features, such as the graph-based location visualisation in LifeSeeker [150]. Other novel interactions such as VR and voice commands, while exciting and promising, are not commonly embraced in the top-3 teams, which are

mostly desktop-based. This is likely due to the lack of support for these platforms and the difficulty of implementing them. However, they remain intriguing and are likely to be explored in the future.

2.2 Approaches to Lifelog Question Answering

Recently, with the rise of cross-modal embedding models, such as CLIP [167] and CoCa [235], large-scale pretrained models have been utilised to extract the visual and textual features from the images and questions, and then rank the images based on the similarity between the features. This embedding-based approach allows a more user-friendly experience by allowing users to search for images using natural language queries and significantly improves the retrieval performance [4, 207]. As a result, most search tasks in the LSC have been solved by the 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.

Before tackling the lifelog QA task, let us first explore the broader landscape of QA. QA may also be considered as an extended version of information retrieval (IR) in which semantic understanding is required to answer the questions [91, 103]. Depending on how much search is involved, QA variations can be broadly categorised into the following types:

- Open-domain QA (OpenQA): This entails answering questions without predefined context. Often, QA systems focus on general factoid questions, such as:
 - ‘*What is the capital of Ireland?*’
 - ‘*What animal is the first to be in space?*’
 - ‘*How high is Mount Everest?*’

In this scenario, the model is required to (1) retrieve the relevant information from a large knowledge base (KB), such as a large crawl of the Web and online encyclopedias, and (2) infer the answers based on the retrieved documents. This is also known as *IR-based QA* [243]. Jurafsky and Martin [91] argued that despite large language models such as GPT-3 [22] and T5 [168] being able to answer general factoid questions, they are prone to *hallucinations*, where the answer is not supported by the evidence. For this reason, OpenQA is still in favour of the two-step architecture mentioned above.

- Comprehension QA: A typical task of this type is Machine Reading Comprehension (MRC), which requires the model to understand the context and answer questions based on it [18]. Less emphasis is placed on the retrieval part, as the context is usually given. To extend this, we can further consider the task of answering questions about

the content of a specific image (Visual Question Answering — VQA) or video (Video Question Answering — VQA) as instances of Comprehension QA.

Lifelog QA presents a novel and relatively uncharted task with no clear guidelines on its approach. It is not yet understood which questions should be asked or which scenario should be considered. This dissertation considers a specific scenario of lifelog QA where the user, who is not the lifelogger, lacks contextual information about the question. This is due to the vast volume of lifelog archives, which makes examining them all at once impractical. Instead, users must search for relevant information and deduce the answer from the retrieved data. As a result, in this work, I classify lifelog QA as an OpenQA task applied in the multimedia domain. In the following sections, I will first discuss existing approaches to OpenQA and Comprehension QA, and then explore the possible approaches to lifelog QA.

2.2.1 Open-Domain Question Answering (OpenQA)

Numerous datasets have been developed to facilitate the development of OpenQA systems, with Wikipedia as a popular source of information, as of the case in HotpotQA [231], Natural Question [107], and SQuAD_{open}[29]. These datasets were designed to test the model’s ability to retrieve relevant information from the entire Wikipedia corpus to answer the questions. The answers to the questions are usually short and can be found in a single document. However, some may require the model to combine information from multiple documents. Most modern OpenQA systems follow a ‘*Retriever-Reader*’ architecture [29, 39, 243] 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. This paradigm is why OpenQA is also known as *IR-based QA* [91, 243]. Figure 2.14 shows the architecture of DrQA [29], a seminal OpenQA system. I will take a look at DrQA in more detail and discuss different approaches to the *Retriever* and *Reader* in this section.

The first part of DrQA, the *Retriever*, is a document retrieval system that uses TF-IDF weighted bag-of-word vectors to compare Wikipedia articles and questions. Bigram counts are incorporated to consider local word order. As the result, five most relevant articles are returned given a question and passed to the *Reader*. Other techniques for the *Retriever*, such as BM25 [230] and more advanced deep retrieval models that encode the question and documents, can also be found in the literature [94, 114, 157].

After retrieving related documents, DrQA’s *Reader* is a neural MRC model that takes the question and the retrieved documents as input and outputs the answer. An RNN model was trained on the SQuAD dataset [170] to predict the start and end positions of the answer span in each paragraph in the retrieved articles. Unnormalized exponential

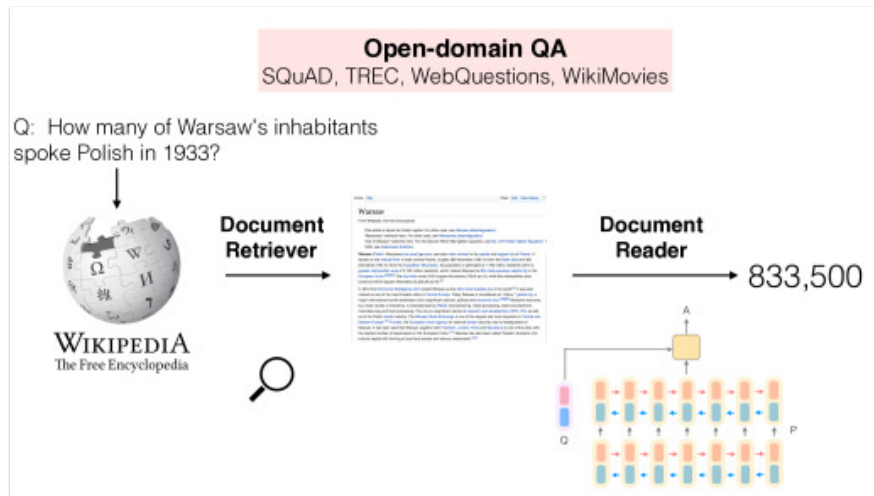


Figure 2.14: The architecture of DrQA [29], a typical IR-based QA system consisting of a Retriever and a Reader.

was used to combine the scores from all paragraphs and the final answer was selected by taking the argmax over all considered paragraph spans. This kind of approach is also known as *extractive* models, which are used in most OpenQA systems [29, 94, 230]. On the other hand, *generative* models [86, 121] applies models such as BART [120] and T5 [168] to generate the answer in an open-ended manner, which is more flexible but less interpretable. Notably, Retrieval-Augmented Generation (RAG) [121] has been proposed to combine the strengths of retrieval-based and generation-based models. RAG’s two-stage architecture resembles the OpenQA paradigm, where the first stage retrieves relevant documents and the second stage generates the answer.

To further extend the architecture, some works [113, 220] proposed re-ranking the retrieved documents before feeding them into the *Reader* [113], or training the entire OpenQA system in an end-to-end manner [114, 121, 157]. An in-depth discussion of these approaches is beyond the scope of this dissertation and we refer to Zhu et al.’s survey [243] for more details.

This OpenQA paradigm is a potential solution to the lifelog QA task. 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. In this research, I take inspiration from the OpenQA architecture and propose steps to adapt it to the lifelog domain, which will be discussed in Chapter 7.

2.2.2 Comprehension Question Answering

In the OpenQA paradigm, after retrieving relevant documents, the task is transformed into a Comprehension QA task, where the model is required to answer the question based on the context of those documents. The context is usually a paragraph or a document in the text domain; thus, Machine Reading Comprehension (MRC) can be used to answer the question. However, in the lifelog domain, the context is usually a set of images or videos, with associated metadata such as time, location, and activity. The metadata can be considered as text and MRC can be used to answer the question. However, to address the visual parts, we need to look at the two closest tasks in the computer vision domain: Visual Question Answering (VQA) and Video Question Answering (VideoQA) tasks. In this section, I will discuss the approaches to these two tasks and how they can be applied to the lifelog domain.

Machine Reading Comprehension (MRC)

Machine Reading Comprehension (MRC) is a task in NLP that aims to equip machines with the ability to read and comprehend textual passages, answering questions posed in natural language [18, 77, 237]. Over the years, MRC has gained significant attention due to its potential applications in information retrieval, question answering systems, and text summarisation. Early MRC approaches primarily focused on rule-based systems and template-based techniques [185]. Recently, with the revolutionary development of deep learning, researchers have proposed many neural network-based MRC models, achieving state-of-the-art performance on many MRC datasets such as SQuAD [170], CNN/DailyMail [78], RACE [108], and MS MARCO [154]. Some of the most seminal works in this area include the Attentive Reader [78], which was based on the attention mechanism of deep neural networks; the Bi-Directional Attention Flow (BIDAF)[190], which was a multi-stage hierarchical process that represents the context at different levels of granularity and uses a bi-directional attention flow mechanism to achieve a query-aware context representation without early summarisation; R-net [221], which incorporated a gating mechanism in attention computation that can dynamically control the model to use information from each part; and FusionNet [85], which fused the word-level embedding and higher-level representations to obtain a more comprehensive representation of the context in a multi-level attention mechanism.

With the introduction of the pre-training model BERT (Bidirectional Encoder Representations from Transformers)[38], the performance of MRC models has been significantly improved. [194, 228] BERT is a language model that was pre-trained on a large corpus of unlabelled text and can be fine-tuned on specific MRC datasets, minimizing the need for extensive task-specific labelled data. This concept of transfer learning has since been extended to models like RoBERTa [127], GPT-2 [166], and more advanced architectures,

leading to state-of-the-art results on multiple MRC benchmarks.

Visual Question Answering (VQA)

Visual Question Answering (VQA) is a bridge between computer vision and natural language processing, challenging AI systems to comprehend images and answer related questions in natural language. VQA has gained significant attention in recent years due to the rapid development of deep learning and the availability of large-scale datasets [92]. The Visual Question Answering (VQA) dataset [11], which contains open-ended questions about images, was one of the first large-scale datasets for VQA. Since then, many other datasets have been introduced, including VQA 2.0 [197], Visual Genome [104]. These datasets vary in terms of the types of questions, the number of questions per image, and the number of images. In order to address the questions in these datasets, various approaches have been proposed.

A common approach of VQA models is to obtain a joint representation of the image and the question in a shared embedding space, and then use this representation to predict the answer. Pretrained CNNs are employed to obtain image representation. Text representations are obtained using recurrent neural networks (RNNs) on pretrained word embeddings. After that, a classifier then predicts the single-word answers from a predefined vocabulary based on the joint representation of the image and the question [172]. Multimodal Compact Bilinear Pooling (MCB)[54] improved on the joint representation by using compact bilinear pooling, which computed the outer product between two vectors, allowing a multiplicative interaction between all elements of both vectors. DualNet [183] integrated both element-wise summations and multiplications to embed the visual and textual features. In addition, attention mechanisms were also used to improve on the above method by allowing interaction between specific regions of the image and the question words [30, 225, 232, 244]. External knowledge bases can also help with the VQA task by providing additional information that is not available in the common visual datasets such as ImageNet [37] or COCO [126].

All these models used a predefined vocabulary to predict the answers, in other words, they treated the VQA task as a classification problem for which the cross-entropy loss is used for training. However, this approach is limited to predicting single-word answers and its robustness is limited by the size of the predefined vocabulary. On the other hand, some viewed the problem as a sequence generation problem, where an encoder-decoder architecture was used to generate the answer free-form [55, 143].

Video Question Answering

Comparing to VQA, Video Question Answering (VideoQA) is a more challenging task as it requires the model to understand the temporal information in videos, which is also a

key feature of lifelog data.

Many datasets have been proposed for benchmarking the VideoQA task in recent years, such as TGIF-QA [87] and TVQA [116], providing video clips with accompanying questions, challenging VideoQA systems to analyse and comprehend dynamic visual scenes. All existing VideoQA datasets except for EgoVQA [50] are from a third-person perspective. TGIF-QA [87] is a dataset of over 165,000 questions on 71,741 animated pictures. Multiple tasks are formulated upon this dataset, including counting the repetitions of the queried action, detecting the transitions of two actions, and image-based QA. MSVD-QA and MSRVT-QA [224] are two datasets with third-person videos. The VideoQA tasks formulated in both of these two datasets are open-ended questions of types what, who, how, when, and where, and their answer sets are of size 1000. YouTube2Text-QA [233] is a dataset of 1987 videos and 122,708 automatically generated QA pairs with both open-ended and multiple-choice tasks of three major question types (what, who, and other). TVQA and TVQA+ [116, 117] are built on 21,793 video clips of 6 popular TV shows with 152.5K human-written QA pairs. EgoVQA [50] was proposed due to the lack of first-person point-of-view videos in these datasets; however, the size of the dataset is small, with just over 600 question-answer pairs.

Most SOTA VideoQA models employed LSTM or GRU-based encoders to encode sampled video frames and question words into sequences of features. These features were fed into a reasoning component to produce the correct answer. Some variants of the attention mechanism were also used to localise and find relevant frames (temporal attention) [87, 141, 224] or regions (spatial attention) [100, 223, 232].

In the manner of most natural language processing problems, the question was transformed into word embeddings using pretrained models such as GloVe 300-D [163], which were then encoded by an LSTM or GRU-based encoder to produce textual features. Video features were extracted from sampled video frames with pretrained neural networks such as ResNet [73], VGG [196] to represent appearance, and C3D (e.g., [87]) to represent motion. Some works [87, 224] adopted early fusion of these features before feeding into the video encoder, while others [51, 56] applied late fusion which is integrated in attention modules. Some of them also integrated a multi-step reasoning approach [51, 233] instead of simply combining video and question features to produce the final answer. Such reasoning modules utilised a controller such as AMU [224] or LSTM [51] which refined the attention over each iteration.

Following the trend of pretraining cross-embedding models, recent research has also explored mass-scaled video-text modeling with contrastive learning for various downstream tasks including VideoQA. For example, VideoCoCa [227] improved on image CoCa and reused attention poolers that are parts of the pretrained image-text model without further retraining. Another work is FrozenBiLM [229] which utilised frozen pretrained visual encoders by integrating lightweight adapter modules to enable zero-shot VideoQA. These

new approaches have achieved state-of-the-art results on various VideoQA datasets, such as MRSVTT-QA [224], ActivityNet-QA [236], and TVQA [116].

2.2.3 Relating to Lifelog Question Answering

As noted before, this research views lifelog QA as a task similar to OpenQA in that the user is not provided with contextual information and must search for relevant information to answer the question. As such, the OpenQA paradigm might offer a viable solution to the lifelog QA task. Lifelog retrieval methodologies discussed in the previous section can be used as the *Retriever* in the OpenQA paradigm. Thus, we only need to focus on the *Reader* part, which is responsible for answering the question based on the retrieved lifelog data.

One of these is its focus on text-based documents, whereas lifelog data is multimodal. Looking at the multimodality of lifelogs and the common practices of lifelog retrieval systems, we can categorise the lifelog data into three main types: (1) visual data (photos and videos), (2) textual data (music listening history, notes, etc.) and metadata that can be considered as text (such as semantic location, activity, and time), and (3) other biometrics data (heart rate, sleep quality, etc.). Since little work has been done on the last type of data for lifelogs, I will focus on the first two types in this section and this dissertation in general.

Most VideoQA techniques extend upon the VQA techniques designed for images. These techniques have revolutionised the comprehension of visual content, allowing systems to interpret images and videos and answer questions related to them. Since visual data is the most prominent in lifelog data, these techniques can be seamlessly integrated into lifelog QA. VideoQA is particularly relevant to lifelog QA, as lifelog data inherently involves a chronological sequence of events, activities, and interactions. By incorporating VideoQA methodologies, lifelog QA systems can effectively reason about the temporal dimension, answering questions about the sequence of actions, changes over time, and the narrative captured in lifelog videos.

On the other hand, textual data or textualised metadata can be incorporated directly into a VideoQA model if such model accommodates *video subtitles*, as in the case of TVQA [116] and FrozenBiLM [229]. However, MRC models offer a more reliable approach to textual data, as they are specifically designed to comprehend textual passages and answer questions based on them. The extractive nature of most MRC models is also an advantage when it comes to location and time data, as it reduces the hallucination as seen in generative models [88] and provides the exact answer.

Dedicating a separate MRC model for textual data and textualised metadata allows longer and more textual context to be considered. However, this approach allows no interaction between the visual and textual data, as opposed to a unified VideoQA model. A unified lifelog QA model that can handle both visual and textual data is the ideal

solution, but it is not yet clear how to achieve this.

Essentially, expertise from the lifelog retrieval domain, as well as MRC, VQA, and VideoQA domains, can be applied to lifelog QA to create a sophisticated framework capable of comprehending textual, visual, and temporal aspects of lifelog data. Because of the novel nature of lifelog QA, a combination of these techniques may be required, pushing the boundaries of lifelog comprehension and setting off in a new era of interactive lifelog retrieval systems that respond to the diverse informational needs of users within their personal lifelogs.

2.3 Conclusion

This chapter has presented a comprehensive review of the literature on lifelog retrieval systems and question answering techniques that are relevant to this research. It discussed the history of lifelog retrieval systems by looking at early systems and reviewed various approaches to solve the task of lifelog retrieval. In summary, the following key guidelines were identified for the development of lifelog retrieval systems:

- **Interactive Retrieval:** The system should support interactive retrieval, which allows the user to provide feedback and refine the search results, leading to more accurate and relevant results.
- **Temporal Context:** Temporal Context was a gap in the literature at the start of this research, but it has been identified as an important feature in lifelog retrieval systems.
- **Support for Novice Users:** Existing lifelog retrieval systems were designed to support expert users, but there was (and still is) a need for systems that are easy to use and provide guidance to novice users.
- **Accuracy and Efficiency:** State-of-the-art techniques in multimedia retrieval should be employed to ensure the accuracy and efficiency of the system.
- **Standard Evaluation:** The annual Lifelog Search Challenge (LSC) has been identified as the main evaluation approach in the community, which has attracted a number of international teams to participate.

Additionally, this chapter identified a critical gap in the literature: the lack of research in the area of lifelog question answering. Following this, a literature review on general question answering techniques was conducted and potential approaches to lifelog question answering were discussed. Specifically, the OpenQA paradigm was found to be a potential solution to the lifelog QA task. The two-stage architecture of OpenQA is flexible enough to incorporate with existing state-of-the-art lifelog retrieval systems without the need to

re-train the entire system. Questions about lifelog data can be answered by adapting the existing state-of-the-art MRC, VQA, and VideoQA models to the lifelog domain.

To address the research objectives in this dissertation, which involve multiple aspects of lifelog retrieval systems, I have decided to follow the common practices in the lifelog retrieval community and approach the research in an incremental manner, starting with the development of an initial lifelog retrieval system, and then extending it to support lifelog question answering.

These factors acted as the main guidelines for this research. In the next chapter, a detailed description of the research methodology and research design will be presented, which will provide a clear understanding of how the research objectives will be achieved.

Chapter 3

Methodology

The research objectives in this dissertation are to develop a lifelog retrieval system with question answering (QA) capabilities and to evaluate the system's performance and usability. Since lifelog retrieval is a relatively new research area that involves a wide range of disciplines, a cycle-based approach is suitable for this research work. By taking part in a multi-cycle process of development, evaluation, and refinement, this research aims to develop a system that engages in meaningful interactions, unlocking the potential of lifelogs to serve as a rich source of insights and knowledge.

Specifically, Design Science Research (DSR) [79] was chosen as the research methodology. DSR is ideal for creating and evaluating innovative systems, artefacts, and methods to address complex problems. It is characterised by its emphasis on improving the functionality and performance of systems through iterative development, evaluation, and refinement. In this case, the artefact is the lifelog retrieval system, and the problem is the difficulty in identifying and retrieving (or answering questions about) relevant information from lifelog data. The iterative and participatory nature of DSR is well-suited to the development of a lifelog retrieval system, as it allows for continuous refinement and enhancement of the system's capabilities. Through these cycles, this work aims to refine different aspects of lifelog QA, adapt to user needs, and address any unforeseen challenges. Finally, after several cycles, the hypothesis is proved or disproved, and the research questions are answered.

In this chapter, I will describe the research design and methodology for developing a lifelog retrieval system with question answering (QA) capabilities based on the principles of DSR. Operating constraints such as time limitations, data availability, and user participation will be acknowledged and addressed. The chapter is structured as follows: Section 3.2 discusses the research design aligned with the principles of DSR, with details about the data, evaluation criteria, participants, and research objectives. Ethics considerations are also discussed in this section. After that, Section 3.3 discusses the operating constraints of the research, and Section 3.4 concludes the chapter.

Table 3.1: Design Science Research Guidelines, borrowed from [79]

Guideline	Description
1 Design as an Artefact	The research must produce a viable artefact in the form of a system, method, or an instantiation.
2 Problem Relevance	The artefact must develop technology-based solutions to important or relevant problems.
3 Design Evaluation	The artefact must be evaluated to demonstrate its utility and effectiveness.
4 Research Contributions	The research must provide clear and verifiable contributions to the knowledge base.
5 Research Rigour	The research must be conducted with rigour and discipline.
6 Design as a Search Process	The research must be a search process, which involves iteration and refinement.
7 Communication of Research	The research must be effectively communicated both to the technical and managerial audiences.

3.1 Design Science Research

According to Hevner et al. [79], Design Science Research (DSR) is one of two primary research paradigms in Information Systems (IS) research, the other being Behavioural Science Research (BSR). Artefacts are the primary outcome of DSR, and they are broadly defined as constructs (vocabulary and symbols), models (abstractions and representations), methods (algorithms and practices), and instantiations (implemented and prototype systems) [217]. Knowledge and understanding are gained through the *building* and *evaluation* of these artefacts. The guidelines for DSR are shown in Table 3.1. Although the guidelines are not strictly followed in this dissertation, they provide a useful framework for conducting DSR.

In this research, the lifelog retrieval system was conceptualised as the core artefact (Guideline 1), motivated by the need to improve the accessibility and usability of lifelog data (Guideline 2). Annual assessments through live benchmarking campaigns and user studies were conducted to measure the system’s utility and effectiveness (Guideline 3). This iterative process of development, evaluation, and refinement was crucial in informing the subsequent actions, demonstrating the search process nature of the research (Guideline 6), and leading to research contributions that extend beyond the system development (Guideline 4). The research rigour was maintained through the use of established evaluation criteria and the documentation of the research process in this dissertation (Guideline 5). Insights and findings were communicated through workshops, presentations, publications, and this dissertation (Guideline 7).

3.2 Research Design

In this section, I will discuss the research objectives, data, evaluation criteria, participants, and ethical considerations of this research. The research design is aligned with the principles of DSR as discussed in the previous section.

3.2.1 Research Objectives

The primary objective of this research is to develop a lifelog retrieval system with question answering (QA) capabilities and to evaluate the system’s performance and usability. Our approach was guided by a series of research questions, addressed through various development and evaluation cycles. While certain cycles were conducted sequentially, others happened concurrently, utilising parallel advancements in different aspects of the system. These overlapping cycles showcased the dynamic and iterative nature of the research process, adapting to the insights gained from each phase and the evolving needs of the system’s users. Figure 3.1 shows the timeline of the research cycles, which are discussed in detail in the following sections.

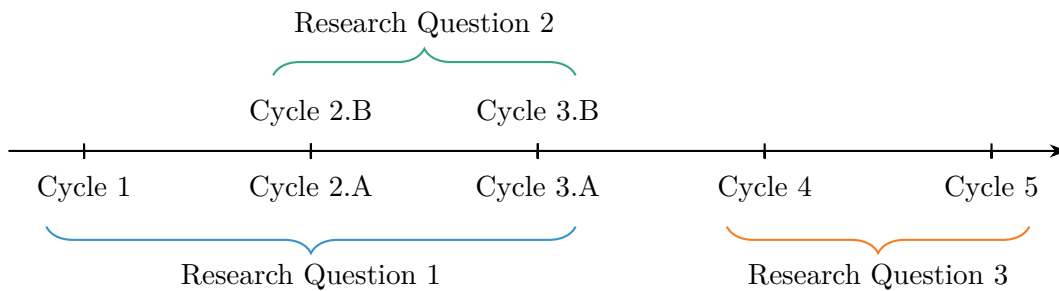


Figure 3.1: Timeline of the research cycles

Cycle 1: Development of the Initial Lifelog Retrieval System

This cycle aimed to develop the initial lifelog retrieval system, which was the baseline system. The system was developed based on the conducted literature review (Chapter 2). The key factors in developing a state-of-the-art system in lifelog retrieval were identified and used to guide the development of the system. This baseline system, which is called Myscéal, was evaluated in the LSC’20 campaign, which is the first cycle of the research. A user study was also conducted to evaluate the usability of the system and to gather feedback for further improvements. This cycle was expected to partially answer research question 1.

Cycle 2.A: Improvement of the Lifelog Retrieval System

This cycle aimed to improve the initial lifelog retrieval system based on the feedback from the first cycle and the development of the lifelog QA dataset. LSC'21 was the first campaign where the improved system was evaluated.

Cycle 2.B: Development of the Lifelog QA Dataset

This cycle happened in parallel with the improvement of the lifelog retrieval system. It aimed to jump-start the use of question answering (QA) in lifelog retrieval by creating a first-of-its-kind lifelog QA dataset. The dataset was created by collecting annotations for lifelog on the LSC'20 dataset mentioned above. A semi-automatic question-answer pair generation method was proposed to create the lifelog QA dataset. This dataset was used to evaluate different QA models for lifelog data. Research question 2A was expected to be addressed in this cycle.

Cycle 3.A: Further Improvement of the Lifelog Retrieval System

Similar to Cycle 2.A, this cycle aimed to further improve the lifelog retrieval system based on the feedback from the second cycle. This system was evaluated in LSC'22, and fully addressed research question 1.

Cycle 3.B: Evaluation of QA techniques for lifelog data

Cycle 3 aimed to compare different QA models on the lifelog QA dataset and choose the best model to be used in the final lifelog QA system. This cycle was expected to address research question 2B.

Cycle 4: First Step Towards Lifelog QA

This cycle aimed to propose a new event-based embedding system and evaluate its effectiveness compared to the baseline system on known-item search tasks. This cycle was expected to address research question 3A.

Cycle 5: Development of the Lifelog QA System

This cycle aimed to build a first prototype of a lifelog QA system, extending the event-based embedding system by adding a QA model. This system was used in the LSC'23 campaign. Moreover, a user study was conducted to evaluate the novice users' interaction with the system and explore more possible questions that can be asked. This cycle was expected to address research question 3B.

While the research questions and objectives were set at the beginning of the research, the research design was flexible enough to allow for changes and adaptations based on the findings from each cycle.

3.2.2 Data

Due to privacy concerns and the effort required to collect, analyse, and annotate a considerable amount of personal data, lifelog collections are not as readily available as other multimedia datasets. To avoid unnecessary complexity and ensure the repeatability of this research, existing lifelog collections that are publicly released by the research community were reused. These archives have already addressed some challenges and guidelines such as: what to log, how often to log, how willing are users to share data, what to share and who can access the data. Specifically, the following lifelog collections were utilised:

LSC'20 Lifelog Collection

The LSC'20 dataset[68] is a four-month multimodal dataset from a single active lifelogger, consisting of data collected from the years 2015, 2016, and 2018. Three files are provided:

- *Core Image Dataset* (38.49GB): 191,439 wearable camera images captured with OMG Autographer and Narrative Clip devices, fully redacted in 1024×768 resolution. Anonymisation has been applied to the images, which means that faces and most readable text have been blurred in a manual or semi-manual process. Private contents that are not suitable for public release have been removed.
- *Metadata* (2.8MB): a CSV file containing the following information for every minute: timestamp, physical activities, biometrics, and semantic locations (for example home, work, Tesco, etc.) of the individual.
- *Visual concepts* (79.9MB): a CSV file containing the visual concepts detected in the non-redacted images. Visual concepts in this file are divided into two categories, which are objects and scenes. The objects are detected using an object detection model trained on the COCO dataset [126]. 80 classes are detected, such as person, car, and dog¹. The scenes are detected using a scene recognition model trained on the Places365 dataset [238]. 102 labels are detected, such as waiting in line, working, and open area². The bounding boxes of the detected objects and confidence scores are provided for each detected concept.

¹For a full list of objects, please refer to https://github.com/amikelive/coco-labels/blob/master/coco-labels-2014_2017.txt.

²For a full list of scenes, please refer to https://github.com/CSAILVision/places365/blob/master/labels_sunattribute.txt

This dataset was also used in ImageCLEF 2020 Lifelog Retrieval Task[155] and LSC'21[69]. The dataset is available on the LSC website, which can be accessed at https://lifelogsearch.org/lsc/2020/lsc_data/.

LSC'22 Lifelog Collection

This dataset is much larger with 18 consecutive months of lifelog data from January 2019 to June 2020, which includes:

- *Core Image Dataset*: 725,000 point-of-view lifelog images captured by a Narrative Clip device, fully redacted in 1024×768 resolution. The same anonymisation process was applied to this dataset in a fully-automated manner. Certain scenes and activities are also removed from the dataset due to privacy concerns.
- *Metadata*: a CSV file containing the following information for every minute: timestamp, physical activities, biometrics. However, semantic locations are not available in this dataset.
- *Visual concepts*: included in the metadata file. The same visual concept detection model is used as in LSC'20 dataset. In addition, Optical Character Recognition (OCR) outputs are also provided for the associated images.
- *Additional Semantic Locations*: a supplementary metadata file, provided by me for the LSC'23 campaign, which used the same dataset. This file contains semantic locations of the lifelogger.
- *Additional flight data*: flight locations as departing airport — arrival airport pairs are provided by the Voxento developer[7], who was also a participant in LSCs.

The LSC'22 dataset is also available on the LSC website, http://lifelogsearch.org/lsc/2022/lsc_data/.

Moreover, in line with earlier studies [125, 226], the most important factors for memory recall and lifelog understanding are the *what*, *where*, and *when* of lifelogging activities. Therefore, this work focused on only these three aspects of lifelog data and excluded biometrics data such as heart rates, step counts, and sleep quality. It is important to note that this exclusion does not eliminate the potential for introducing the remaining data at a later stage. However, due to time limitations and the research's scope, incorporating these elements within this research was not feasible.

We can see some example lifelog data from both datasets in Figure 3.2 and Table 3.2. The images in both datasets are redacted, which means that the faces and most of the text are blurred, as seen in image (B) in Figure 3.2. The redaction process, although necessary, sets a trade-off between privacy and the ability to understand the context of the lifelog. Quite often, the redaction process also removes important visual cues. Some other issues

regarding lifelog images are also shown in the figure, such as blurry images (C), images with very small details that require zooming in (D), and images with very similar visual content that are hard to distinguish and take up visual space (E). The text in (C) are not blurred because they are not considered private. These are some limitations of the community dataset, and it is important to be aware of this when considering the results of the research.



Figure 3.2: Example lifelog images from both datasets.

On the other hand, Table 3.2 shows some rows from the provided metadata files. The metadata files contain a variety of data, such as timestamps, GPS coordinates, semantic locations, visual concepts, and biometrics. However, since my focus is not on the biometrics data, I exclude heart rates, step counts, and sleep quality from the metadata files. Moreover, the visual concepts provided by the organisation were not used in this dissertation, as I used my own visual concept detection models. As shown in the table, a minute ID is assigned to each minute of lifelog data, and oftentimes multiple images are captured in the same minute. Moreover, the semantic locations, such as Work and Home, are not available in the LSC’22 dataset, which is one of its limitations.

Another limitation of these datasets is their geographical bias. Although the lifelogger had made an effort to capture a wide range of activities and locations, a large portion of the data was captured in Dublin, Ireland. As an attempt to mitigate this bias, all the queries used in the LSC campaigns were designed to be location-agnostic and avoiding giving an advantage to the users who live in the same area as the lifelogger. Given the constraints of the research, the geographical bias was not addressed in this dissertation. However, this bias to some extent limits the generalisability of the research results.

Table 3.2: Selected non-visual metadata from both datasets. Visual concepts are not shown here due to space limitations and the fact that this dissertation did not make use of the provided visual concepts, but rather employed my own visual concept detection models.

Dataset	minute_id	utc_time	latitude	longitude	semantic_name	ImageID
LSC'20	20150226_1226	2015-02-26 12:26:00	53.3854	-6.2571	Work	b00000569_21i6bq_20150226_122554e.jpg, b00000570_21i6bq_20150226_122632e.jpg
	20150301_2146	2015-03-01 21:46:00	53.3892	-6.1582	Home	b00001482_21i6bq_20150301_214550e.jpg, b00001483_21i6bq_20150301_214620e.jpg
LSC'22	20190103_2021	2019-01-03 20:21:00	53.3759	-6.1925	Raheny Service Station, Howth Road, Dublin, County Dublin, D05 A0K1, Ireland	20190103_202127_000.jpg, 20190103_202159_000.jpg
LSC'22	20190220_1320	2019-02-20 13:20:03	53.3861	-6.2575	Collins Avenue Extension, Dublin, County Dublin, 9, Ireland	20190220_132626_000.jpg, 20190220_132658_000.jpg

3.2.3 Ethical Considerations of Lifelog Data

As lifelog data is a form of personal data, understanding and addressing the ethical implications of this research is crucial. The lifelog data utilised in this study, while publicly available, still contains information that is inherently personal. Recognising the potential for privacy violations, I have taken steps to ensure that the data is used responsibly and ethically as follows:

- **Ethics Approval:** Ethics approval was obtained from the Ethics Committee of Dublin City University, which is the institution where the research was conducted. The approval reference number is DCUREC/2023/127.
- **LSC Terms and Conditions:** The data used in this research was used in accordance with the terms and conditions set by the LSC organisers. The data was used for research purposes only and was not shared with any third parties. User studies were done in controlled environments and the data was not shared with the participants. The systems developed in this research were not made publicly available, but only for authorised researchers to access.
- **Secure Data Storage:** The data was stored on secured servers and was only accessible to the researchers involved in the project. Proper measures were taken to ensure that the data was not accessible to unauthorised individuals. Note that the stored data was not the original data, but rather the processed data after the redaction process.
- **Future Use of Data:** The data used in this research will not be used for any other purposes. The data will be securely stored and will be deleted after the completion of my PhD studies.

Whether lifelogging has a future as a mainstream practice is still uncertain. However, the potential for lifelog data to be used in a variety of applications, such as health monitoring, memory augmentation, and personal assistance, is clear. Responsible and ethical use of lifelog data is crucial to ensure that the potential benefits of lifelogging are realised without compromising the privacy and security of individuals.

3.2.4 Evaluation Criteria

Along with the datasets, a decision was made to use the queries and evaluation criteria from established benchmarking campaigns. This was to ensure that the research's outcomes were comparable to the state-of-the-art and to facilitate the evaluation process. Specifically, the works in this dissertation were used to participate in the annual Lifelog

Search Challenges (LSC) [68–71] and a side task in ImageCLEF 2020 [155]. Chapter 2 introduced the metrics used in these campaigns and the following are the evaluation criteria chosen for this research:

Interactive Scoring Metrics

It is important to note that the LSC is an *interactive* benchmarking campaign, which means that user interaction is a key component of the evaluation process. As mentioned before, the LSC uses a live scoring system which allows the user of each participating system to submit their queries and receive the judgement in real-time. For known-item search (KIS), the evaluation metrics are based on the time taken to submit the correct image and the number of wrong submissions. The score is calculated as follows:

$$\text{score} = 100 - 50 \times \frac{\text{time taken}}{\text{time limit}} - 10 \times \text{number of wrong submissions} \quad (3.1)$$

If the task is not solved, the score is 0. The penalty for wrong submissions is set at 10 points for all the LSC campaigns. The time limit for KIS is 300 seconds (5 minutes). This scoring system is designed to encourage the participants to submit the correct image as quickly as possible and to avoid submitting wrong images. The scoring system is also used in the Video Browser Showdown (VBS) [76]. Using this scoring scheme, the user is encouraged to wait for more evidence before making a submission, since the penalty for wrong submissions is much higher than the reward for a quick submission. Furthermore, this minimises the impact of the system’s processing speed on the score, as such delays are insignificant compared to the time limit.

Regarding the question answering (QA) task, which is the focus of this dissertation, this scoring system is employed with some modifications. In LSC’21, since the task was formulated as a known-item search, the scoring is the same as the KIS task, except that (1) the time limit is 180 seconds (3 minutes) instead of 300 seconds and (2) only one submission is allowed. In LSC’22, the scoring is modified to allow multiple text-based submissions and human judges are needed to evaluate the correctness of the submitted answer. Then, the formula in Equation 3.1 is used. The use of human judges here is crucial because there might be multiple acceptable answers due to the ambiguity that the questions might have.

For ad-hoc queries, the evaluation metrics are based on a pooled set of relevant documents for each query, the number of relevant documents retrieved, and the number of wrong submissions. Human judges will judge the relevance of the submitted images in real time and the score is calculated as follows in Equation 3.2:

$$\text{score} = 100 \times \frac{\text{correct}}{\text{correct} + \text{incorrect}/2} \times \frac{\text{correct}}{\text{total}} \quad (3.2)$$

Here, *correct* is the number of relevant images submitted, *incorrect* is the number of irrelevant images submitted, and *total* is the total number of relevant images in the groundtruth pool. The number of incorrect images is divided by 2 to reduce the penalty for wrong submissions because the number of submissions in the task is unlimited. Similarly, if no relevant images are submitted, the score is 0.

The two mentioned metrics are standard in the multimedia retrieval community and also used in video retrieval benchmarks such as the Video Browser Showdown (VBS) [76].

In the LMRT ImageCLEF challenge, *precision* and *recall* are used to evaluate the ad-hoc retrieval task. Precision, also known as positive predictive value, is the fraction of retrieved instances that are relevant, while recall, also known as sensitivity, is the fraction of relevant instances that are retrieved. Table 3.3 and Equation 3.3 show the calculation of precision and recall based on the number of relevant and irrelevant images retrieved and not retrieved. Generally speaking, precision indicates how relevant a retrieved item is, while recall indicates how many relevant items are retrieved in a search. These metrics are traditionally used in information retrieval tasks, where the relevant instances are documents and the retrieved instances are the results of a search. ‘Documents’ in this case are images. Note that images can be very similar to each other if they were captured at a relatively short time interval. In such cases, each image is considered as a separate instance, which might not be intuitive to the user. This is a limitation of the evaluation metrics used in the LMRT ImageCLEF challenge.

Table 3.3: Precision and Recall calculation

	Relevant	Irrelevant
Retrieved	True Positives (TP)	False Positives (FP)
Not Retrieved	False Negatives (FN)	True Negatives (TN)

$$\text{Precision} = \frac{TP}{TP + FP} \quad \text{Recall} = \frac{TP}{TP + FN} \quad (3.3)$$

F1 measure is the harmonic mean of precision and recall, as shown in Equation 3.4.

$$\text{F1-measure} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3.4)$$

Automatic Retrieval Metrics

This section is more in line with the traditional information retrieval evaluation metrics, which are used to evaluate the performance of the system without user interaction. In order to choose the best approaches for the backend of the interactive retrieval system, the Hit Rate at Rank K (H@K) was considered as one of the evaluation metrics. If the top K results contain at least one relevant result, then H@K is 1, otherwise, it is 0. This

is appropriate for the scoring metrics in interactive LSCs, where the user is only required to submit one correct image. As a result, if one of the top K results is relevant ($H@K = 1$), the user will most likely be able to identify and make a correct submission.

3.2.5 Users

In this dissertation, I am interested in three kinds of users: the lifelogger, the expert, and the novice users. The lifelogger is unique in that they are the ones who generate the lifelog data. In this context, there is only one lifelogger for both LSC'20 and LSC'22 datasets. Ideally, the lifelogger is the most important target user for the lifelog retrieval system. However, due to the limited availability of the lifelogger, I focused on only the expert and novice users as with all other systems.

Expert users are those who have a great knowledge of lifelog retrieval systems and lifelog data. In the live LSC campaigns, often times expert users are the system developers themselves. Measuring the performance of the expert users is important to show the full potential of the system.

On the other hand, novice users have little to no experience with lifelog retrieval systems or even the lifelog concept. They are also the target users as their feedback is crucial. Even though they lack the knowledge of someone else's lifelog, it would be more difficult for them to find the relevant lifelog events as they did not experience the events themselves. However, testing lifeloggers is not feasible due to privacy concerns and the rarity of lifeloggers even in the research community. Therefore, novice users are suitable if the system is designed to be used by the general public.

In this dissertation, I analysed the expert performances at the live campaigns and conducted user studies with novice users to evaluate the usability of the system. Although the results from the expert users do not directly reflect the general public's performance, they are more likely to identify the problems in the system compared to the novice users [184].

In the user studies, to recruit participants, convenience sampling was used. That means I recruited participants that I had access to, such as my social contacts and colleagues, from a variety of disciplines. This is a type of non-probability sampling, which is often used in exploratory research. In this case, the researcher is interested in getting a fast and reliable approximation of the truth. One of the main advantages of convenience sampling is that it is quick to implement and cost-effective. However, the main disadvantage is that the sample may not be representative of the population, which limits the generalisability of the results. For instance, the participants in the user study might have a higher level of education, a higher interest in technology, or of a younger age than the general population. However, since I am only interested in the usability of the system, I believe that this sampling method is sufficient for my research.

3.3 Operating Constraints

As with any new topic of research, I have clearly defined the constraints within which the research operated. I have identified the constraints thus:

- Time was a critical constraint, shaping the pace and scope of this research. Given an estimated four-year period for the research, there was a finite time frame for completing each research cycle.
- As no new lifelog data is collected, the performance of the system is constrained by the quality of the chosen lifelog collections. The constraints related to data availability, data diversity, and data quality might have influenced the system’s performance and its ability to retrieve relevant images and generate accurate answers. Efforts were made to work with representative lifelog datasets while acknowledging potential data limitations.
- Annotation efforts were constrained by the availability of volunteers who were willing and suitable to annotate the lifelog data. As a first step to explore the potential of lifelog QA, there were no guidelines on what were the best annotations to be collected. The quality of the annotations was also constrained by the volunteers’ ability to understand the lifelog data.
- Limited types of questions were generated for the lifelog QA dataset. I focused only on yes/no and multiple-choice questions for ease of evaluation. The format of the questions was also influenced by third-party question generation tools. The size of the dataset was also limited due to the lack of assistance and the time constraint.
- Similarly, the availability of state-of-the-art algorithms and models, as well as the computational resources to run or train them, had a significant impact on the development of the lifelog retrieval systems. This research adhered to available resources, as opposed to developing new algorithms, while still achieving system functionality and performance.
- The willingness and availability of participants to contribute, test the system, and offer valuable feedback posed a significant constraint on the generalisation of the research findings. The research was conducted with a limited number of participants, and the results may not be generalisable to the broader population.
- Balancing functionality, ease of use, and aesthetics within the constraints of available design tools and expertise was a consideration that influenced the system’s usability and user experience.

- Developing new QA approaches in the information retrieval domain was not the focus of this research. Instead, state-of-the-art QA models were adapted to the multimedia and lifelog domains.

These constraints were maintained for this Ph.D. research and acted as limiting factors to focus the research effort and scope.

3.4 Conclusion

In conclusion, this chapter described the research design and methodology for developing a lifelog retrieval system with question answering (QA) capabilities based on the principles of Design Science Research. The research objectives, data, evaluation criteria, participants, and ethical considerations were discussed. The research design was aligned with the principles of DSR, which emphasises the development and evaluation of an artefact, the search process nature of the research, and the communication of research findings. The research was conducted in a series of iterative and participatory cycles, which allowed for continuous refinement and enhancement of the system's capabilities. Operating constraints such as time limitations, data availability, and user participation were acknowledged and addressed. The next chapter will discuss the baseline lifelog retrieval system, which is the starting point of this research.

Chapter 4

Interactive Lifelog Retrieval

The work in this chapter is in collaboration with my research team, therefore, the pronoun *we* is used throughout the chapter. My role as the primary contributor to this work was to design the system, implement the core backend, develop the user interface, use the system to participate in the Lifelog Search Challenges, and conduct user studies.

This chapter focuses on Research Question 1, which is **How to design a state-of-the-art interactive lifelog retrieval system that assists a novice user to quickly locate items of interest from a conventional multimodal lifelog?**

To address this question, we developed a state-of-the-art interactive lifelog retrieval system called **MyScéal** based on the literature review of state-of-the-art retrieval systems in Chapter 2. MyScéal incorporated most of the conventional techniques found in other lifelog retrieval systems and has been improved through three research cycles.

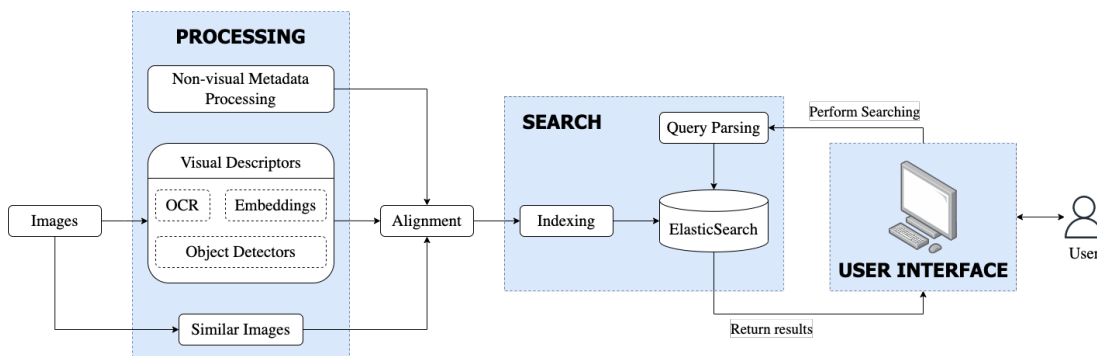


Figure 4.1: Pipeline of MyScéal.

MyScéal was firstly designed as a concept-based interactive lifelog retrieval and later upgraded to an embedding-based system to improve free-text search. Nevertheless, the pipeline of the system remained the same, which is illustrated in Figure 4.1. Visual concepts extracted from the images and activity descriptors, GPS coordinates, and semantic locations were used to create an inverted index in the ElasticSearch engine. User interactions were transformed into ElasticSearch queries. The user interface was designed to

Table 4.1: Comparison of different versions of Myscéal.

Component	MySceal	MySceal-CLEF	MySceal 2.0	E-MySceal
Visual processing	aTF-IDF, DeepLabv3+, Microsoft Vision	aTF-IDF, DeepLabv3+, Microsoft Vision	aTF-IDF, DeepLabv3+, Microsoft Vision, Google API OCR, Material and colour concepts	Google API OCR, CLIP ViT-L/14@336px
Data Processing				
Location data	-			GPS clustering
Time data		Time-of-day, day, month, year, etc.		
Primary Ranking Algorithms		aTF-IDF, TF-IDF		Cosine similarity
Visual Similarity		SIFT & VGG16		CLIP
Temporal search		Before — Main — After search		
GPS search		Map filtering		
Query Expansion		Wordnet + ConceptNet		-
Search boxes		Before — Main — After search		
Query suggestion/feedback	-	Exposed word expansion, Manual modification	Exposed word expansion	-
Search results	Triplet	Images	Triplet	Adaptive
Map	Showing result clusters	-	Showing result clusters, Showing relevant locations	
Visual Similarity		Horizontal Scrolling		Timeline

present the results in a straightforward manner, which allows the user to quickly select and submit the targeted image to the evaluation server. All key features of different versions of Myscéal are summarised in Table 4.1. In this chapter, I will discuss each feature in detail and explain our decision for the updates. The chapter is organised as follows: the data processing components are described in Section 4.1, the search process is presented in Section 4.2, and the interactive aspect of lifelog systems is addressed in Section 4.3. Next, Myscéal’s performance is addressed in Section 4.4. Finally, I conclude the chapter in Section 4.6 by discussing the key features of a state-of-the-art interactive lifelog retrieval system.

4.1 Data Processing

As discussed in Chapter 2, the data processing component is the most important part of a lifelog retrieval system. In this section, I will discuss the data processing components of Myscéal in detail. As seen in Figure 4.1, the data processing component consists of three main parts: visual descriptors, and non-visual metadata processing. Image embeddings are also extracted for similar images and their similarity scores are calculated during the processing stage to speed up the search process.

4.1.1 Visual descriptors

Following the standard approach of enriching visual concepts of many systems in previous years, we utilised additional features from existing state-of-the-art computer vision models as follows:

- Semantic labels from DeepLabv3+ [31], an image semantic segmentation model. This process labels each pixel in an image with corresponding visual concepts, resulting in a segmentation map.
- Object detection and image captioning concepts from Microsoft Computer Vision API ¹,
- Optical Character Recognition (OCR) results from Google Cloud Vision API ².
- Material and colour concepts from Bottom-up attention model [9] trained on Visual Genome Dataset [104].

Furthermore, Myscéal exploits the *area* of each visual object in an image, or in other words, the pixel count in the semantic segmentation result and the bounding box area in the object detection result³. The idea behind this is the assumption that bigger objects

¹<https://azure.microsoft.com/en-us/services/cognitive-services/computer-vision/>

²<https://cloud.google.com/vision>

³Here, we also consider an OCR text as a visual object with its bounding box.

play a more important role in an image. This information is used to in our scoring scheme called **aTFIDF**[205]. This is one of my *main contributions* as aTFIDF was introduced in the first version of Myscéal.

The following is the detailed explanation of this scoring scheme. For a given image, if we denote the set of images from the lifelog dataset by I , the collection of possible object keywords by O , the area of an object detected in that image by $f_{o,i}$, $o \in O$, $i \in I$, we can calculate the **area-term frequency** as following:

$$aTF(o, i) = 1 + \log(f_{o,i})$$

The area-inverse document frequency can be obtained by the following:

$$aIDF(o) = \log\left(\frac{N}{\|\{i \in I : f_{o,i} > c\}\|}\right)$$

where

- N : total number of images in the dataset
- c : a constant that is used as a threshold for the area for determining if an object is actually in the image or if it is visual noise. This has been set empirically as 10% of the image area.

Finally, the **aTFIDF** can be calculated as follows:

$$aTFIDF(o, i) = aTF(o, i) * aIDF(o)$$

In E-Myscéal following the advancement of text-image embedding models, we utilised a pretrained image-text embedding model, Contrastive Language-Image Pre-training (CLIP) ViT/L-14@336px [167] to replace the aforementioned visual concepts. Subsequently, this increased the performance of the system tremendously, which in turn reduced the user's time and effort required to solve a query.

4.1.2 Non-visual metadata processing

The lifelog datasets used by the LSCs consist of images and associated non-visual metadata, namely, GPS coordinates, time in UTC, as well as (misaligned) local time and activity recognition from biometrics data (walking, transport, etc.). Amongst these, Myscéal did not use other data like music or heart rate, as this information did not contribute much to the previous LSC events. The two most important factors were time and GPS coordinates.

Regarding time data, we converted UTC time to local time based on the time zone detected from the GPS coordinates. The days of the week (Monday, Tuesday, etc.) were

also extracted from time information.

Regarding GPS coordinates, in all versions, location data were used as a main factor in segmenting the lifelog into meaningful units. In other words, we segmented the lifelog using a hierarchy where the longest unit’s boundaries are defined by a change of location (or more specifically, the *semantic names* of location).



Figure 4.2: GPS clustering in Myscéal 2.0.

In the first two versions of Myscéal we directly utilised the official semantic names for segmentation. There were two main issues with this approach. First, even in the LSC’20 dataset, the semantic names were not always available. Second, in LSC’22, the organisers did not provide the semantic names, which is realistic in real-world scenarios.

Addressing the first issue, we used GPS clustering to enhance the location data. In Myscéal 2.0, we employed clustering methods to find a centre point for data points that share identical semantic names as seen in Figure 4.2. If a data point whose semantic name was not available is within a certain distance of a cluster centre, it would be assigned the same location name as that cluster centre. This was done to support the user interaction with a map, which is discussed in Section 4.3. Depending on the locations mentioned in the query, we chose relevant cluster centres to visualise on the map. For example, if the user was searching for ‘*Eating lunch in Dublin City University*’, the centre of all data points associated with the semantic name ‘*Dublin City University*’ would be presented on the map with the location name.

The second issue posed a much bigger challenge. In E-Myscéal, we continued to use GPS clustering to enhance the location data. For each cluster, OpenStreetMap API⁴ was used to extract addresses of the data points within the cluster. Then, the most common address was assigned to the cluster centre. Semantic names in this case were the first part of the address. For instance, if the most common address was ‘*Dublin City University*,

⁴<https://www.openstreetmap.org/>

Glasnevin, Dublin, Ireland, the semantic name would be *Dublin City University*. OpenStreetMap API was also used to extract the names of states and countries to support the search process. All location-related texts were then normalised to ASCII to avoid any encoding issues.

However, the problem remained that GPS coordinates were not always available, especially indoors. This is not limited to the LSC datasets, but is widespread in various location-based research [20]. Further improvements in lifelog location data are necessary. After LSC’22, I had the opportunity work with the organisers to address this issue for LSC’23. We achieved this by incorporating visual information to infer the location of the images, as discussed in VAISL [210], although this is beyond the scope of this thesis.

4.1.3 Temporal units

In Myscéal we defined a temporal hierarchy of events consisting of three units: **image**, **scene**, and **event**. The smallest temporal unit was an **image**, which is an atomic unit of a lifelog in many works. We considered a **scene** to be the combination of one or many subsequent similar *images*. An example is when the lifelogger is working at a desk, and his surroundings remain practically unchanged. An **event** consists of one or multiple consecutive *scenes*, whose boundaries are indicated either by a change of contexts, such as location and activity, or by a significant time gap. These units were used throughout indexing, searching, and user interaction.

We segmented lifelog into *events* using location semantic names and activities. To define the segmentation boundary of *scenes*, we assigned each image three feature vectors including VGG16 feature [196], Word2Vec feature [146], and SIFT feature [136, 137]. We compared adjacent images by calculating three cosine distances and building a Naive-Bayes classifier to determine scene boundaries using these distances. In E-Myscéal, we found that using a simple threshold on the CLIP ViT/L-14@336px [167] embeddings was sufficient to determine scene boundaries.

4.2 Search Process

As seen in the pipeline in Figure 4.1, the search process consists of two main steps: indexing, query parsing, and searching using the Elasticsearch engine. The indexing step is to create an inverted index of the data, which is used to filter the data based on non-visual information. The query processing step is to transform the user’s input into a query that can be used to retrieve the data from the index.

4.2.1 Elasticsearch indexing

We employed an off-the-shelf search engine called Elasticsearch⁵ to index the data. Elasticsearch, an open-source search and analytic engine, supports searching with varied data types. The lifelog index was created as a collection of JSON-like documents with the properties shown in Table 4.2 in the `scene` index. This database provided a quick way to *filter* the data based on non-visual information, as seen in Table 4.2, and was used as a way to narrow down the search space before more complex calculations were applied to each image. The main `image` index was created with more fine-grain properties, as seen in Table 4.3. This index was used to *score* the images based on the query. The scoring process is discussed in Section 4.2.3.

Table 4.2: Elasticsearch document for each scene.

	Explanation	ES Format	Examples
<code>images</code>	the list of image IDs	keyword	20160810_071508_000, 20160810_071421_000
<code>begin_time</code>	local time	date	2016/08/10 08:12:00+00
<code>end_time</code>	local time	date	2016/08/10 08:12:00+00
<code>desc</code>	the list of visual concepts with equal importance	keyword	station, red wall
<code>weekday</code>	the day of the week	keyword	monday
<code>location</code>	provided semantic name of the location	keyword	home, angelica's cafe
<code>address</code>	reverse geocoding result	text	whitehall, dublin, ireland
<code>gps</code>	GPS coordinates	geo_point	53.3858, -6.2607
<code>activity</code>	provided activity recognition	keyword	transport

4.2.2 Query parsing

We decided not to use a faceted interface to show filters in different metadata such as date, time, and location to reduce the number of actions that the user has to take to interact with the system. Our reasoning was twofold: (1) this saves the user's time and effort in selecting the filters, and (2) the user might not be familiar with the metadata of the lifelog data to make the correct actions. Therefore, we chose to use a single text box for the user to enter the query. The purpose of the query parsing process is to transform the user's input into a query that can be used to retrieve the data from the Elasticsearch index.

⁵<https://www.elastic.co/>

Table 4.3: Properties of Elasticsearch document for each image. For visual concepts that lack areas, we use the Elasticsearch keyword data type and configure Elasticsearch to calculate the TF-IDF scores. *: only available in E-Myscéal.

	Explanation	ES Format	Examples
<code>image_id</code>	the image ID	keyword	20160810_071508_000
<code>time</code>	local time	date	2016/08/10 08:12:00+00
<code>atfidf_s</code>	aTFIDF feature from semantic segmentation	rank_features	{"wall": 1.35, "person": 6.79}
<code>atfidf_o</code>	aTFIDF feature from object detection	rank_features	{"wall": 1.35, "person": 6.79}
<code>atfidf_ocr</code>	OCR feature	rank_features	{"online": 16.892, "book": 18.00}
<code>concepts</code>	visual concepts that lack areas and thus aTFIDF scores	keyword	["wall", "person"]
<code>feat*</code>	CLIP feature	dense_vector	[0.1, 0.2, \ldots]

Using ad-hoc regular expression patterns, we mapped the textual query into corresponding fields.

Before E-Myscéal, with the concept-based approach as the main search mechanism, we also extracted a list of visually descriptive words from the query. After that, we used Word2vec [146] and WordNet [160] to map the words into a specific and limited set of keywords provided by object detectors and semantic segmentation engines. For example, *tea* might imply the presence of a *mug*, or a *teapot*. This process transformed every concept into a list of keywords with the corresponding relevance scores. In E-Myscéal, the direct mapping from the query (minus date, time, and location information) to the CLIP embeddings was used instead.

4.2.3 Primary search mechanism

For a processed query, time-related information (day of the week, date, month, year, or time of the day), locations of large areas (cities and countries), and activities were used as filters (that is exact match). Semantic locations were used as GPS filters based on the result of GPS clustering on top of pure text matching, based on the assumption of incomplete annotation.

In the first three versions of Myscéal, the system used a concept-based approach that relies on TF-IDF and aTFIDF scores. The system first applied the metadata filters to the `scene` index. To further narrow down the search space, the system scored the remaining scenes and took the top N scenes. This scoring process is as follows:

For a list of visual concepts, $q = [q_0, q_1, \dots, q_m]$, where each concept q_i is expanded into a list of keywords $[o_{i,0}, o_{i,1}, \dots, o_{i,n}]$, $o_{i,j} \in O$ with the relevance scores of $[r_{i,0}, r_{i,1}, \dots, r_{i,n}]$, the final score of each scene can be formulated as:

$$S_{\text{scene}} = \sum_{i=0}^m \max_{\{j|o_{i,j} \in \text{scene}\}} r_{i,j}$$

After that, using the images belonging to these scenes, we add the scores of each image based on (1) the relevance score in the field `concepts` which was calculated using TF-IDF, and (2) the aTFIDF scores in the three fields `atfidf_s`, `atfidf_o`, and `atfidf_ocr` in Table 4.3. Each aTFIDF score is calculated as follows:

$$S_{\text{image}} = \sum_{i=0}^m \max_j (r_{i,j} * \text{aTFIDF}_{\text{image}}(o_{i,j}))$$

In E-Myscéal, we moved away from the concept-based approach and used the CLIP embeddings directly. We employed `Elasticknn`⁶, a plugin for `ElasticSearch`, to support nearest neighbour search. This plugin allows us to use the cosine distance as a scoring function and to retrieve the top N images. Specifically, given a query embedding q and an image embedding i , the score is calculated as follows:

$$S_{\text{image}} = \text{cosine_similarity}(q, i) = \frac{q \cdot i}{\|q\| \|i\|}$$

4.2.4 Complementary search mechanisms

Aside from using a single textual input as the search query, Myscéal offered other means of search.

Temporal search After analysing the tasks from previous LSCs, we decided to support users to search for multiple time-related events. One example task is from the LSC'20 [68]: *‘Just looking at coffee machines one evening in a luxury store in Dublin called “Brown Thomas”. I had just finished a long meeting under two lights before walking to Brown Thomas. Afterwards I had a birthday dinner in Sole restaurant. It was the last day of May in 2018.’* Three different activities were mentioned in the query:

- The main activity: *‘looking at coffee machines one evening in a luxury store in Dublin called “Brown Thomas”’;*
- The activity before: *‘I had just finished a long meeting under two lights before walking to Brown Thomas’;*
- The activity after: *‘I had a birthday dinner in Sole restaurant’.*

⁶<https://elastiknn.com/>

It is clear that the user was looking for images of the main activity, but the system should also return images of the activities before and after to provide a temporal context. Therefore, we designed a temporal search mechanism that allowed the user to specify the temporal relationships between different queries. Specifically, the system would then search for the main query first and then use the resulting time information as a time filter to search for the other events. All results would be grouped at the last step and ranked based on their total score.

Let us look at an example with two temporal queries: ‘*I was eating lunch in DCU before I went to the airport*’. The system first searched for ‘*eating lunch in DCU*’, resulting in a list of scenes called $M = [M_0, M_1, \dots, M_m]$. Then, for each scene M_i in M , its time information was extracted and added to the query ‘*went to the airport*’, forming a new query such as ‘*went to the airport after 12:00 on 23/08/2016*’, resulting in a list of scenes called $C_i = [C_{i0}, C_{i1}, \dots, C_{in}]$. Combining M with C resulted in pairs of scenes (M_i, C_{ij}) , which were then scored based on the sum score of M_i and C_{ij} .

It is worth nothing that Myscéal was the first system to support multiple temporal queries. While being a complex search mechanism, this feature proved to be very useful in many tasks and was gradually adopted by other systems in the LSCs [4, 97, 132], as mentioned in Chapter 2.

GPS search The search query could also be extended with a location filter using a bounding box of GPS coordinates drawn on the user interface. In the case of temporal queries, the filter was only applied on the main query.

Visual similarity search Visual similarity can help the user find visually similar images to any given image by clicking on it in the search result. The similarity scores were calculated using the cosine distance of a concatenation of the SIFT [136, 137, 139] and VGG16 [196] features. CLIP embeddings were used in E-Myscéal instead.

4.3 User Interaction

The final part of the pipeline, as illustrated by Figure 4.1, is the user interface. The user interface is the most important part of an interactive lifelog retrieval system where the user interacts with the system. In this section, I will discuss the user interface of Myscéal in detail. The user interface of Myscéal was designed to be simple and intuitive, allowing the user to quickly interact with the system. The user interface can be divided into four main parts: the search boxes, the query suggestion, the search results, and the map. The user interface of Myscéal is shown in Figure 4.3. Apart from that, a pop-up window called *Event View* is used to show the images in a temporal context. *Similarity View* is also used to show visually similar images to the selected image. We will discuss the UI components in detail in the following sections.

Due to the expectation of novice users’ involvement, the UI was designed with two

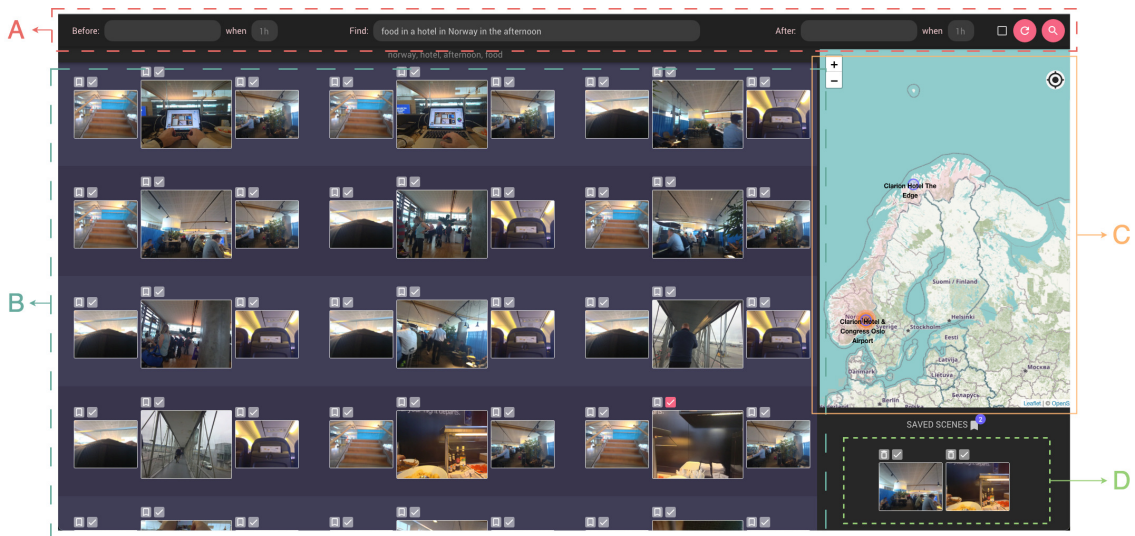


Figure 4.3: User interface of Myscéal. The search can be initiated by filling in the search boxes (A) and pressing the *Enter* key on the keyboard. The search results are shown on the left side of the screen (B) in a list of triplets. The map on the right side of the screen (C) shows the locations of the images in the search results. The bottom right corner of the screen shows the saved section (D), which allows the user to save the search results for later use.

main principles: to minimise different steps in the search process and allow back-end functionality to be fully used without requiring lifelog search expertise.

4.3.1 Search boxes

The three boxes at the top of the UI were to specify the temporal relationships between different query clues. The primary query was placed in the middle to allow for quick entry. Furthermore, as discussed in Section 4.2, the system supported multiple temporal queries such as ‘*I was in Dublin City University before I went to the airport*’. Therefore, in the small boxes, the time conditions (in hours) of “before” and “after” queries could be specified.

4.3.2 Query suggestion

Since the second version of Myscéal, we have exposed the query expansion to help the user adjust the query accordingly. The second version [208] allowed the user to modify the relevance score of each visual concept or remove the concept altogether. However, this feature was removed in later versions. In the Myscéal 2.0, we showed the interpretation of the query under the search boxes and highlighted if a word did not appear in the indexed database, prompting the user to double-check that word and select another option in the suggested list if necessary. Since E-Myscéal, we have moved away from the concept-based

approach and used the CLIP embeddings directly. Therefore, the query suggestion was no longer needed.

4.3.3 Search results display

A large proportion of the screen was dedicated to displaying the search results. Each result unit was arranged corresponding to the temporal relationship, with the main event to be searched for in the middle as seen in Figure 4.3. As we segmented lifelog data into *scenes*, each thumbnail here represented a *scene* consisting of multiple images. By clicking on the thumbnail and opening Event View (as illustrated in Figure 4.4), the user could access all images belonging to the selected scene.



Figure 4.4: Event View window in Myscéal, Myscéal-CLEF, and Myscéal-2.0.

In E-Myscéal we adopted an adaptive way of showing the search results based on the number of temporal queries. In other words, if a ‘before’ or ‘after’ query was specified, the system would show the results in pairs or triplets, with the scenes corresponding to the main query in a bigger size. Otherwise, the results would be shown in single images to allow more images to be shown on the screen.

As ImageCLEF challenge focuses on recall performance, we did not group the images into scenes in the search results. Instead, we organised the results into day units and sorted the days based on the maximum score of the images on that day, as seen in Figure 4.5. This allows the user to submit as many images as possible to the evaluation server, as this functionality was optimal to the ImageCLEFLifelog’s evaluation protocol.

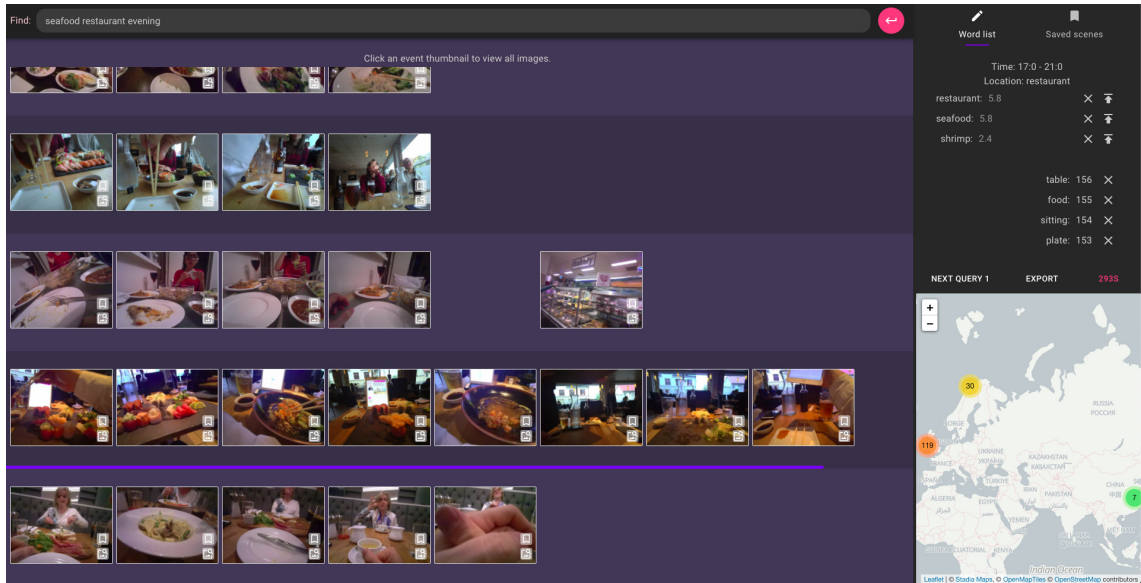


Figure 4.5: User interface of Myscéal in ImageCLEF 2020

4.3.4 Map

Another part of the UI was a map, as seen in Figure 4.3. By default, it showed the locations of the retrieval result. However, it also supported GPS search in an area by allowing the user to draw a rectangle on the map specifying the intended area. Location names, including semantic locations such as home or workplace related to the query, were also shown on an overlay layer of the map.

4.3.5 Visual Similarity and Event View

This view was a pop-up panel that can be accessed by clicking on any image shown in the main interface in Figure 4.3. The purpose of this view was to allow the user to explore the lifelog images in a temporal context. By pressing the ESC key on the keyboard, the user could close the pop-up window and go back to the initial search result screen.

Before E-Myscéal, the event view presented the hierarchy of the three temporal units mentioned in Section 4.1.3, allowing the user to browse the images from that day at two different paces, as seen in Figure 4.4. The first row showed the images belonging to the selected scene, and the second row showed the scenes belonging to the same day. The last row groups the scenes into events, which are the largest temporal units.

Moreover, a user could search for similar images by clicking on a small visual similarity icon at the bottom of each photo in the Event View. The Visual Similarity view was of the same design as the Event View, with the chosen image in the first row, and the similar images (grouped by scenes) in the second row.

In response to user confusion stemming from the design we just described, where two

different functionalities (Event View and Similarity View) were presented in the same design in the same pop-up window, we have redesigned the Event View in E-Myscéal, as depicted in Figure 4.6. Within this pop-up window, users could then *explore temporally-nearby scenes*. Users had the capability to navigate through scenes that were temporally adjacent to the selected scene on the chosen day, with the selected scene highlighted in red. This feature proved particularly valuable in the context of lifelog retrieval, as it allows users to explore scenes that are closely related in time to the scene of interest. These nearby scenes were neatly organised in vertical groups and labelled based on the embedding-based modeling, as described in the previous section, with a helpful textual guideline on the vertical timeline in Figure 4.4. On the other hand, users could also *explore other visually-similar scenes* to the selected scene, which are scenes that share visual similarities with the selected scene, drawn from the entire lifelog archive. This served as a complementary retrieval method while users were navigating through scenes arranged chronologically above, increasing the likelihood of identifying relevant scenes.

On the right side of the pop-up window (to the left of the geographical map), a vertical panel presents enlarged images associated with the selected scene, facilitating navigation within a scene that contains multiple images. Hovering the mouse cursor over the small icon in the bottom right corner of each image triggers further magnification, allowing more detailed inspections, consistent with the behavior of images in all other panels, including the within-day scene list panel, the ‘similar scenes’ panel, and the initial search result screen. Magnification is necessary in cases where the relevant details are too small to be seen in photo thumbnails, such as text on a sign or a book cover.

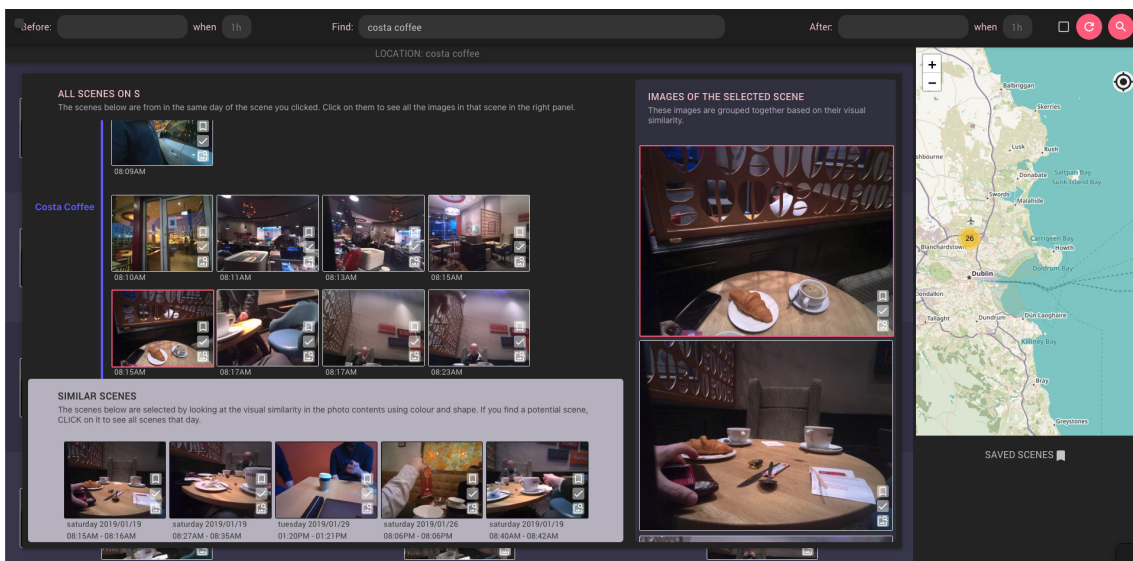


Figure 4.6: Event View window in E-Myscéal, merged with the Visual Similarity view.

4.4 Performance

In this section, we highlight some results that Myscéal achieved at various challenges and how the system performed in different settings.

4.4.1 Myscéal at LSC’20

LSC’20 was the third edition of the LSC series, and also the first lifelog retrieval challenge that Myscéal participated in. A total of 24 tasks were provided to the participants. Despite being the new retrieval system in the competition, Myscéal obtained the highest overall score among 14 participants and achieved first place in LSC’20. Figure 4.7 illustrates the precision and recall of all teams in the challenge, in which the order of teams indicates their final ranking. The precision is defined as the portion of correct submissions out of the total submissions in the competition. Meanwhile, recall is the percentage of tasks that a team manages to solve successfully. We can observe in Figure 4.7 that Myscéal and SOMHunter [145] got the highest recall compared to others at 87.5% meaning that both systems managed to solve 21/24 tasks. Moreover, Myscéal also had the highest precision metric at 84%, which indicates that the system only submitted a few wrong answers with four incorrect submissions out of 25 submissions in total during the competition.

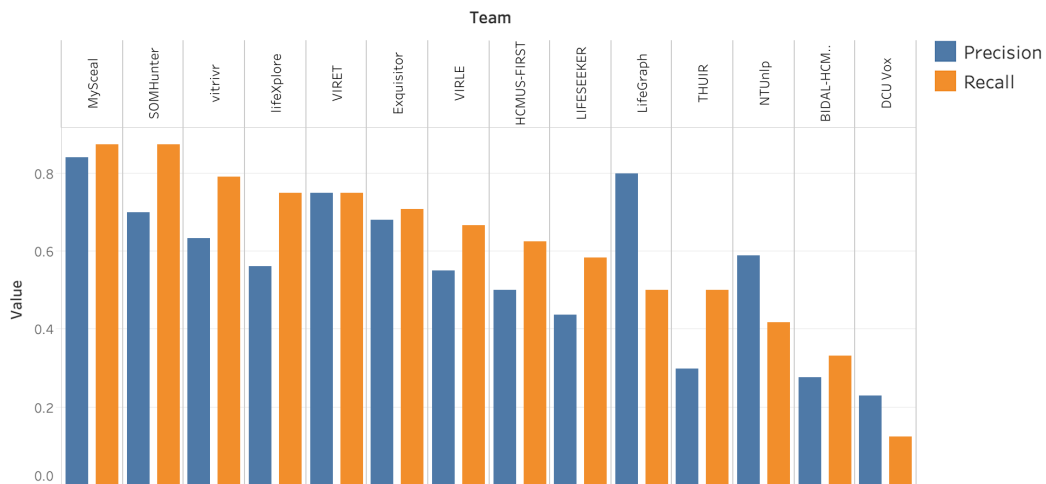


Figure 4.7: Precision and Recall of each team in LSC’20.

The LSC also evaluates systems by taking into account the speed of submission (faster is better). Figure 4.8 depicts the retrieval speed of participants in LSC’20. Myscéal was in the top three quickest systems to return the correct result 13 times, which was the highest across all participants. This means that half of the time in the competition (13/24) Myscéal found the correct answers in the top-3 fastest systems. This search time criterion is one of the key factors helping Myscéal obtain the first place within the LSC’20.

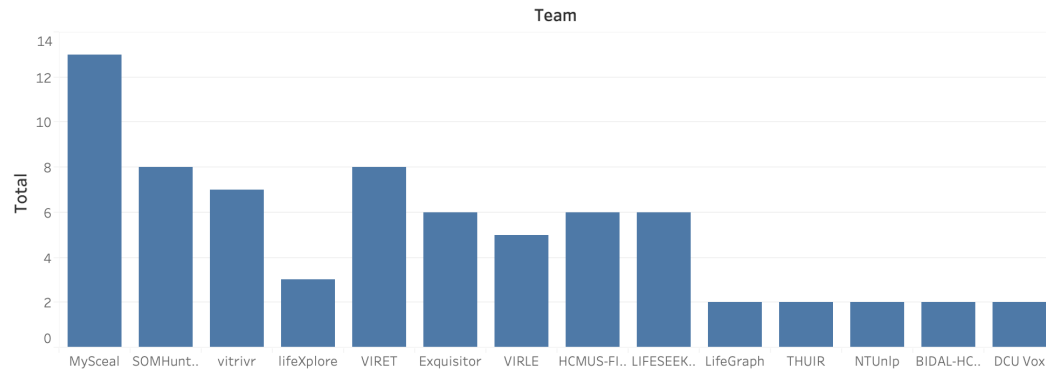


Figure 4.8: Number of times each team was in the top 3 quickest teams to return the correct results.

Due to the COVID pandemic, LSC’20 was the first time the challenge was organised virtually. Therefore, the challenge only had a session for experts and did not have a session for novice users as it had facilitated in previous years. We conducted a user study using Myscéal with eight novice participants to have more insights into the system’s performance when used by novice users. Furthermore, their feedback was a valuable resource from which we can build a better system for LSC’21 with additional features. The setting of the experiments was replicated from the live campaign regarding the time limit and the order in which the clues are shown. However, only five tasks of varying difficulty were chosen from the official list of tasks used in LSC’20 to keep the experiment sessions short. These tasks are illustrated in Table 4.4. They were arranged in an ascending level of difficulty based on our experience in the official competition. Before doing the experiment, all novice users were briefly instructed to learn how to operate Myscéal by trying to solve three sample tasks in a pre-experiment session. These three tasks were also selected from the query bank of the LSC’20.

The score of each participant is shown in Table 4.5, which was calculated using the same formula from the live event. The score of Myscéal is the score gained by expert users (both members of the Myscéal team) in the actual LSC’20 event. As can be observed in Table 4.5, all users (including experts) fail to find the correct answer to the last task. The first two tasks were easier to solve compared to others when all searchers managed to find the relevant images. As we ordered tasks based on their difficulty, task 1 was the task that users obtained the highest score across all tasks. This is because of the keyword “*airplane*” when there was only a small amount of images containing airplanes within the dataset. Meanwhile, in the second task, the participants could not find the correct answer until the last clue of location, “*UK*”, was revealed. As a result, they could manage to get a considerably low score. This also happened in the third task with many images containing beer bottles in Wuhan, making users unable to find the correct answer. In contrast, the

Table 4.4: List of tasks used in our novice user study. The tasks were chosen from the query bank used in LSC’20. The symbol ‘/’ separates clues that are gradually revealed to searchers.

Task	Clues
1	Taking a photograph of an A380 airplane/ in Germany/ before boarding a flight/ in the late afternoon/ in 2015/ on the 19 th March.
2	It was the best cake I had in years,/ in an antiques store./ I was alone drinking tea and eating cake./ I think I finished all the cake in 3 minutes./ It was in the UK/ on a Saturday morning.
3	I was having beer after a long day of meetings./ It was a ‘corona extra’ beer in a bottle./ I remember the room was dark./ I was relaxing in a hotel lobby bar./ I don’t remember there was anyone else there./ It was in May 2018, in Wuhan.
4	Passing by a clocktower while running/ in a park near my home./ It was in the early morning, around dawn./ I drove to the park/ and I drove home again afterwards./ It was a Saturday morning in February.
5	Four red figures,/ maybe they are aliens./ It looked like a painting of aliens./ There were walking on the desert./ There was a big red wall behind the painting./ And a TV, I think there was also a TV there./ I was having tea and sandwiches in March 2015.

location “*home*” in the second hint of the fourth task allowed users to have enough time to find the right images. Six novice users solved three tasks, and two users (U6 and U8) got two correct results. This indicates that novice users could use the system without significant issues since even the expert could only find four correct answers. None of the eight novice users could solve the last task, which is as expected since the expert could not make it either in the official competition. This was because Myscéal in LSC’20 could not detect the color in images, and the word “*aliens*” represented a significant semantic and lexical gap between the information need and the dataset. There was still a big gap in the performance between novice and expert users when the average score of novice users was 188.97, which is just half of the score obtained by the expert user at 339.03.

The most significant issue of Myscéal used in the LSC’20 challenge was that this system did not implement colour detection. Therefore, none of the users managed to find the answer to Task 5 in Table 4.4 where the clue about “*red wall*” was the most informative hint. The same was true for the OCR clues because Myscéal did not have the OCR feature. On the other hand, Task 3 in Table 4.4 could be easily solved if Myscéal had the feature of searching for text to find “*corona extra*”. Additionally, we found that users tended not to

Table 4.5: Experiment score of eight novice users compared to Myscéal team’s official score in LSC’20.

Task	Myscéal	U1	U2	U3	U4	U5	U6	U7	U8
1	94.86	92.5	94.86	85	86.81	98.19	87.92	90.69	82.36
2	78.06	54.86	50.69	68.61	47.08	59.03	57.92	43.33	59.58
3	87.5	53.47	0	0	53.33	0	0	0	0
4	78.61	0	77.5	51.11	0	52.5	0	64.44	0
5	0	0	0	0	0	0	0	0	0
Total	339.03	200.83	223.06	204.72	187.22	209.72	145.83	198.47	141.94

use the map area in the UI although the clues about location contained useful information. We overcome these problems by adding three major updates to Myscéal to participate in LSC’21 [206] and LSC’22 [207]. We included the colour detector and the OCR in the annotation processing component. Furthermore, we also enlarged the map area in the UI to encourage users to use this unique feature.

4.4.2 Myscéal-CLEF at ImageCLEFlifelog’20

Myscéal was developed to match the evaluating criteria of the LSC, which requires a system to find a single specific image that is relevant to a semantic query as quickly as possible with the least number of wrong submissions. ImageCLEFlifelog provides another opportunity to enhance Myscéal as this competition is a more conventional asynchronous retrieval challenge. It requires participating systems to find as many relevant images as possible and does not take the retrieval time into account. We slightly modified Myscéal from LSC’20 by adding an event row at the bottom of the UI as shown in Figure 4.4. This feature was expected to help users scroll faster to find all relevant images. Moreover, we added a small feature that could help users adjust the scores of input keywords to revise the results [208]. Despite not originally matching the challenge’s objectives, Myscéal obtained third place in ImageCLEFlifelog’20 [208].

This competition is also a good opportunity to evaluate Myscéal performance in different use cases, and we conducted additional user experiments with three users: an expert, a novice user, and the data owner (the lifelogger). It is essential to point out that ImageCLEFlifelog is a more suitable challenge than the LSC to include the lifelogger as a user without worrying about their prior knowledge about the dataset. This is because the LSC only needs a lifelog image indicating a specific moment to solve a query, although there are maybe many of them that match the query. This makes it easy for the lifelogger who owns the data to solve, as they can recall the most recent event relevant to the query. However, ImageCLEFlifelog requires a searcher to retrieve all images instead of one. This

means that the lifelogger needs to remember all events, which makes it more difficult for them to solve the query if they use only their memory without using any lifelog retrieval system. For example, to get the maximum points for the query “*Find the moment when the lifelogger was getting a bus to their office*” in ImageCLEFlifelog, the searcher has to find all images belonging to the relevant moments which might happen many times in different days.

ImageCLEFlifelog’20 contained 10 queries as tasks to be solved⁷. For each task, searchers need to find the top 100 images belonging to all relevant moments matched with the corresponding query. In our experiment, each of the three users had a total of five minutes to solve a task, reading time not included. The lifelogger and the novice user were quickly instructed to learn how to use Myscéal prior to the experiments. We used the same evaluation metric in ImageCLEFlifelog’20, which is the F1@10 score. In order to get the highest F1@10 score (which is 1) of a task, the top 10 images of the result should belong to all events described by the query.

Table 4.6: F1@10 scores of three users (U1: Lifelogger, U2: Expert, U3: Novice). The symbol ‘—’ indicates that the user was unable to find the answer for that task. The numbers with * are the highest number in that topic.

Task	U1 (lifelogger)	U2 (expert)	U3 (novice)
1	0.58	1*	0.67
2	0.72*	0.22	—
3	1*	0.57	1*
4	0.31*	0.22	—
5	0.68*	0.68*	—
6	0.25	0.5*	0.25
7	0.69	0.89*	0.69
8	0.75	1*	—
9	0.8*	0.73	0.77
10	0.5	0.5	0.5
Overall	0.63*	0.63*	0.39

Table 4.6 shows the score of all users in the experiments. We can see that the expert, who had the advantage of knowing the system and being familiar with most of the dataset,

⁷<https://www.imageclef.org/system/files/ImageCLEF2020-test-topics.pdf>

achieved the highest score. Despite having no experience with the system, the data owner obtained comparable scores in most tasks and got the same overall score. The average $F1@10$ score of the novice user was lower than that of the others due to the fact that this user was unsuccessful in solving nearly half of the tasks in the challenge. Additionally, although having knowledge of the dataset, the lifelogger got three tasks with the highest $F1@10$ score, which was lower than that of the expert at 4.

Although the lifelogger and the expert user successfully solved all tasks, the novice user only found the answer for half of them. This raised the question of how effective the Myscéal interface was for novice users. Having horizontal scrolling to browse the images in the same row, on top of the normal vertical direction, could be confusing. We observed that both the lifelogger and the novice user rarely used this feature. Furthermore, the implemented keyword scoring adjustment feature was not as helpful as expected when both users completely ignored this function. It did not contribute much to the result of the expert user when the revised result after modifying the weights was not relevant to the queries. Therefore, we decided to remove this feature from Myscéal in LSC'21 [206] and LSC'22 [207].

4.4.3 Myscéal 2.0 at LSC'21

Myscéal participated in LSC'21 with additional features and updates in the user interface, which were based on comments and feedback from novice users in our user study described in Section 4.4.1 and 4.4.2. Specifically, more visual concepts were added to the system with the colour detector and the OCR. The map area was enlarged to encourage users to use this feature, with the relevant locations highlighted on the map. Moreover, the word expansion mechanism was shown in the UI to help users choose the correct word for their query.

Like the previous iteration, LSC'21 was a virtual competition, hence could not have a novice session. There were 23 tasks used in LSC'21, roughly similar to LSC'20 at 24 tasks. The number of participating systems increased from 14 to 17 in LSC'21. The other settings of the competition remained the same as in its previous campaign.

Myscéal obtained the first place in LSC'21 as it did in the previous year. However, LSC'21 witnessed a more competitive performance between the top-ranked teams, when differences in the scores of the top-3 systems were minuscule. Summary scores of the top-6 systems in LSC'21 are illustrated in Table 4.7 in which precision and recall are defined as discussed in Section 4.4.1.

The metric Submitted in Top-3 indicates how fast the systems performed. This is the number of times that a system manages to be in the top speediest systems to find the correct answer.

The overall score is the normalisation of the total score awarded by solving the tasks in the competition. This normalised score is the main metric used to rank the systems

Table 4.7: Summary of LSC’21 result of top-6 systems. The numbers in bold are the highest numbers among the top 6 systems. Precision and recall are defined in Section 4.4.1.

	MySceal	SomHunter+[132]	LifeSeeker [150]	Voxento [6]	CVHunter [131]	Memento [3]
Solved tasks	19	19	20	18	15	16
Wrong submission	4	9	6	3	8	11
Precision (%)	82.61	67.85	76.92	85.71	65.21	59.25
Recall (%)	82.61	82.61	86.95	78.26	65.21	69.56
Submitted in Top-3	12	12	9	11	5	9
Overall Score	100	97.6	97	91.4	77.3	77.2

in the LSC. Table 4.7 shows that there is a negligible gap in the scores of Myscéal, SomHunter+ [132], and Lifeseeker [150]. Although Myscéal attained the best overall score, Lifeseeker was the team that solved the most tasks (20/23) in the challenge and got the highest recall at 86.95%. Regarding precision, Myscéal was not the best in this metric either when Voxento only had three wrong submissions, making it the highest precision at 85.71%. Myscéal, with 19/23 successfully solved tasks and three incorrect submissions, had the same precision as with recall at 82.61% for both metrics. It is interesting that Myscéal was not the system that solved the most number of tasks nor had the least wrong submission, yet managed to rank highest in the challenge. This is because Myscéal was one of the fastest systems that could find the correct answer compared to others. As shown in Table 4.7, Myscéal and SomHunter+ were the two systems that had the highest times submitting the correct answer in the top-3 fastest systems in the competition with 12 times.

Some of the essential features of Myscéal in LSC’21 were OCR and colour detection. Half of the tasks in LSC’21 included the OCR clues from which participating systems could easily find the correct answer. Furthermore, we used a similar image search function many times in LSC’21 to find the relevant result from the initial result. Therefore, we continuously integrate these features for LSC’22, but there are some changes in the UI where we make the similar images panel easier for users to access and explore. Another critical update was that we changed the approach of Myscéal for LSC’22 [207]. LSC’21 witnessed the effectiveness of embedding techniques when SOMHunter+, Voxento, and Memento quickly solved tasks describing activities that were difficult for Myscéal to find

answers. Therefore, instead of using keywords as in previous versions, we change Myscéal to E-Myscéal[207] which applies an embedding approach to participate in LSC'22.

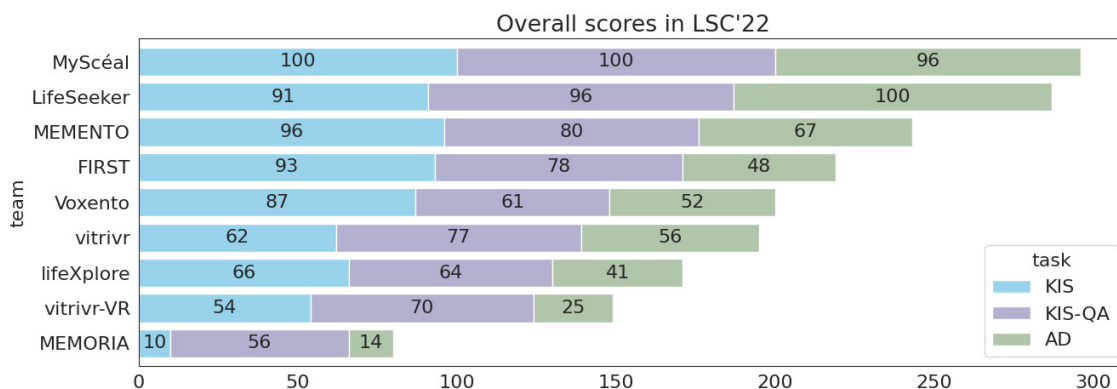
4.4.4 E-Myscéal at LSC'22

The embedding technique used in E-Myscéal is expected to improve novice users' retrieval experience. Unlike Myscéal, the new search mechanism in E-Myscéal does not require users to modify their query several times before entering it into the search bar. Our non-faceted user interface has also been updated to remove components that are potentially confusing to new users, such as the query suggestion. On the other hand, the Event View and Visual Similarity View have been merged into a single pop-up window to reduce the number of pop-up windows that novice users have to deal with.

The LSC'22 was organised in a hybrid manner with some teams participating in person at the conference venue, while others, including E-Myscéal joined remotely. No novice session was held in LSC'22. In this iteration, on top of the usual known-item search (KIS) tasks, two additional kinds of tasks were introduced: Ad-hoc and Question Answering (QA). In total, there were 10 KIS, nine QA, and six Ad-hoc tasks. The QA tasks in LSC'22 were formulated as a KIS task with a question as the query, and the answer was the image that contained the answer to the question. However, only one submission was allowed for each QA task. To avoid confusion with the real QA task in LSC'23 (where text-based answers were required), we will refer to this task as KIS-QA in this paper.

Figure 4.9 shows the overall score of all teams in LSC'22. Same as previous years, E-Myscéal once again scored the highest overall score in LSC'22. However, the differences between that and the score of the second-place team, LifeSeeker [151], were not significant. E-Myscéal ranked first in KIS and KIS-QA tasks and second in Ad-hoc tasks, while LifeSeeker ranked first in Ad-hoc tasks.

Figure 4.9: Overall score of all teams in LSC'22.



Regarding the number of solved KIS queries, half of the teams managed to solve all 10 KIS tasks as Figure 4.10 shows. E-Myscéal shared the same accomplishment with

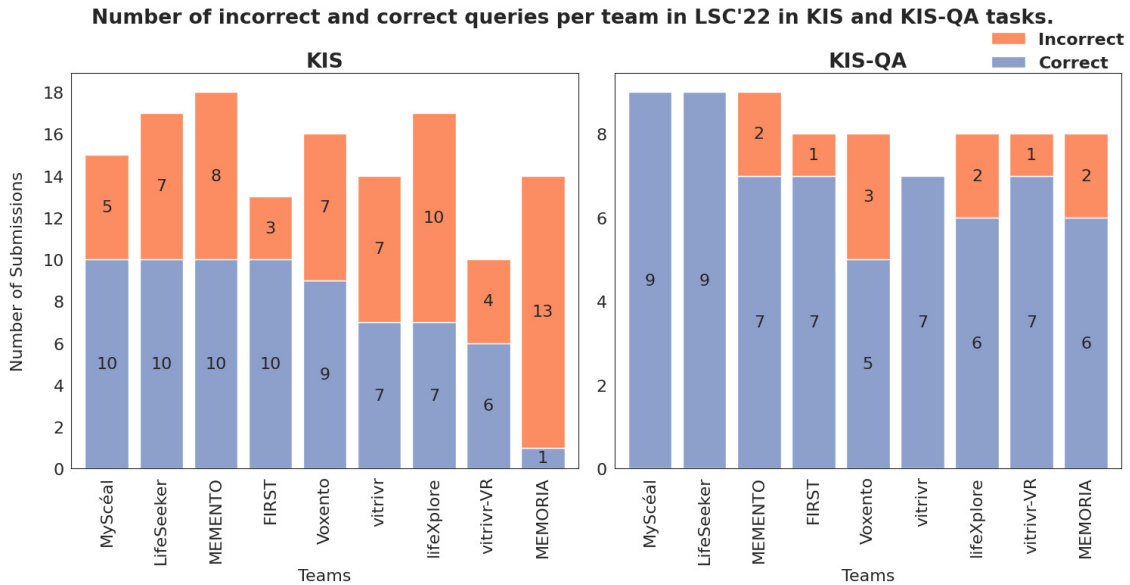


Figure 4.10: Number of incorrect and correct submissions of E-Myscéal in LSC'22.

five incorrect submissions in this category. Although FIRST [80] was the team with the lowest number of incorrect submissions (three incorrect submissions) in KIS tasks, the overall score of E-Myscéal was still the highest due to the speed of solving the tasks, as indicated in Figure 4.11. As we can see from the figure, E-Myscéal was one of the fastest systems to solve the KIS tasks. Similarly, E-Myscéal demonstrated proficiency in KIS-QA tasks, achieving a perfect precision and recall of 1.0 for solving nine tasks. This excellent performance was shared with Lifeseeker [151], yet E-Myscéal's speed in task completion granted it an edge. With respect to Ad-hoc queries, as reflected in Figure 4.12, E-Myscéal achieved the second-highest scores in both precision and recall, with 0.83 and 0.44, respectively. Despite having a much higher precision than LifeSeeker [151] (0.7), E-Myscéal still lost to LifeSeeker in this category due to the lower recall (0.44 compared to 0.54). Overall, the second place in Ad-hoc tasks was an accomplishment for E-Myscéal.

4.5 Discussion

Myscéal was originally developed for the LSC competition, which is to quickly find a single image that is relevant to a semantic query. The result of Myscéal in LSC'20 has shown the system's efficacy with the powerful search engine and the straightforward user interface. Both compartments have helped Myscéal to win the LSC campaigns in three consecutive years: LSC'20, LSC'21, and LSC'22, by solving most of the tasks faster than other teams. Hence, in 2020, Research Question 1, *How to design a state-of-the-art interactive lifelog retrieval system that assists a novice user to quickly locate items of interest from a conventional multimodal lifelog?*, was answered by Myscéal when this system represented the

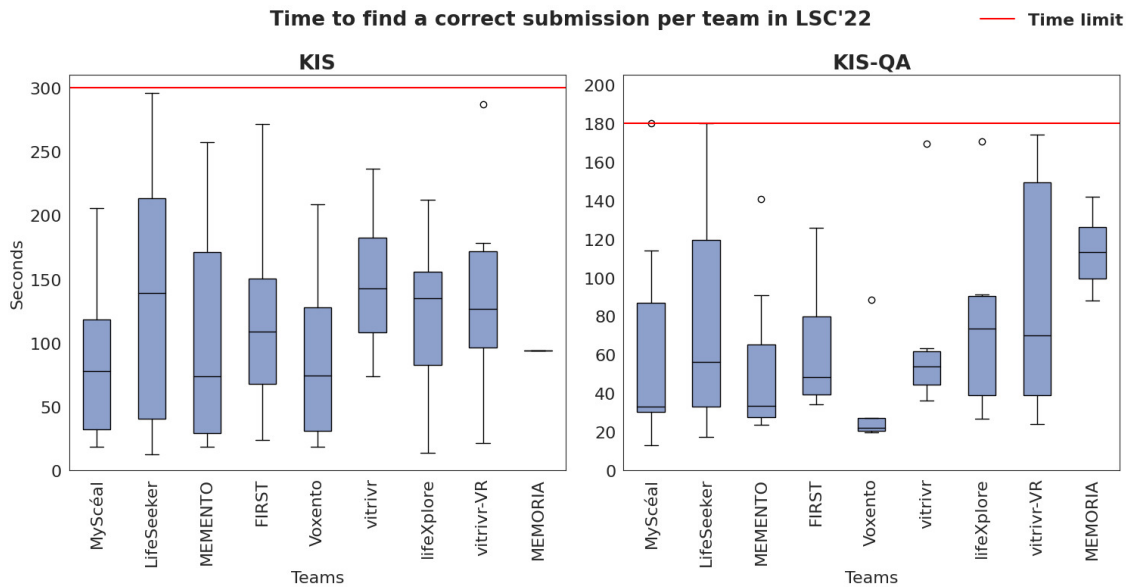


Figure 4.11: Time to first correct submission of different teams in LSC’22. Ad-hoc tasks are not included as they are not scored based on time.

state-of-the-art lifelog retrieval system.

Nevertheless, although Myscéal surpassed other systems in terms of precision, recall, and search time by a large margin in LSC’20, Myscéal achieved first place in LSC’21 with a tiny difference to other teams when this system did not perform significantly better in any metrics. In addition, LSC’21 witnessed a rise in the number of tasks that contain hints about the visible text in the answer images, with 11/23 tasks having OCR information. These OCR clues play a critical role in helping the systems find the correct images, as most of the time teams solved tasks based on them. Across 12 times that Myscéal was one of the three fastest teams, there were eight times that Myscéal found the answers using the OCR feature, which was only implemented for LSC’21. This OCR update indeed came from the feedback of novice users in our experiments (Section 4.4.1) when they commented that it would be easy to solve Task 3 in Table 4.4 if they could use OCR to search for “*corona extra*”. In addition, we also had some modifications in Myscéal after LSC’20 to prepare for LSC’21 based on our observations in the user experiments. For example, the map area in our interface was then enlarged to effectively grasp the attention of users since most novice users did not use this helpful feature as they did not realise there was a map in the interface. However, Myscéal remained the state-of-the-art lifelog retrieval system in LSC’21.

Regarding ImageCLEFlifelog’20, since Myscéal was not created to match the evaluation metrics of this challenge, the system could only achieve third place in this competition. However, we considered ImageCLEFlifelog’20 to be a good opportunity to conduct a user study, including the lifelogger as a user for Myscéal. Table 4.6 showed that the expert user

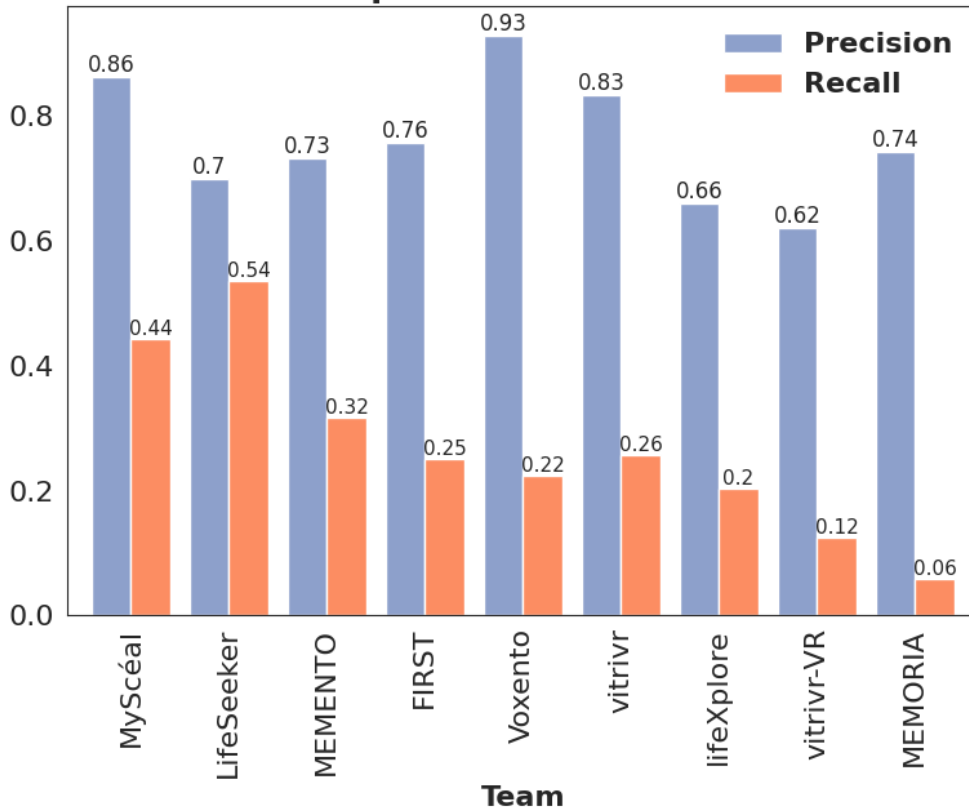
Precision and Recall per team in LSC'22 for Ad-hoc tasks.

Figure 4.12: Precision and Recall of the Ad-hoc tasks in LSC'22.

could have a similar score to the lifelogger with significant knowledge about the dataset. However, although knowing the dataset can have an impact on the lifelog retrieval result, this merit is not enough to gain a high score in ImageCLEFlifelog'20 when this competition required searchers to find all relevant moments. This is because these events sometimes cannot be remembered by the data owner due to the massive size of the dataset with nearly 200.000 images. The target of a lifelog retrieval system is the lifelogger since the system is just a tool supporting them to recall a specific event. Having the same scores for both the expert and the lifelogger shows the benefits of using Myscéal to retrieve lifelog images since the expert does not know the dataset as well as the lifelogger but understands how the system works. Furthermore, we believe that if the lifelogger, who is already familiar with their own dataset, has enough time to learn how to use Myscéal, they can even achieve a better score.

For E-Myscéal we shifted from a keyword-based approach to an embedding-based approach. This is because we have observed the effectiveness of the embedding technique in LSC'21 when SOMHunter+, Voxento, and Memento quickly solved tasks describing activities that were difficult for Myscéal to find answers. E-Myscéal in LSC'22 has shown

the competitive performance of the embedding-based approach when this system solved all KIS and KIS-QA tasks. However, E-Myscéal was not the best system for Ad-hoc queries, which is a similar task to ImageCLEFlifelog'20, as we lacked a good method for fast submissions that was similar to one used by LifeSeeker [151]. Several improvements can be made to E-Myscéal to improve its performance in Ad-hoc tasks. For example, filtering out images that are already submitted can help E-Myscéal to avoid submitting the same images multiple times and save time for other submissions, therefore increasing the recall of this system. In addition, a relevance feedback mechanism can be implemented to suggest images that are similar to the (correctly) judged submissions. This can help E-Myscéal to find more relevant images that are not in the initial result. Nevertheless, E-Myscéal still achieved first place in LSC'22 and represented the state-of-the-art lifelog retrieval system in 2022.

4.6 Conclusion

I have described Myscéal, which was the state-of-the-art lifelog retrieval systems from 2020 to 2022. Some user experiments have been discussed to offer insights into the system's performance. Myscéal applied the conventional and standard approach in this research field, which was to annotate images using its visual concepts but introducing our own new feature called aTFIDF. This novel feature was introduced with the belief that larger visual objects in an image will be more important than smaller objects. In addition to the search engine, Myscéal comes with a clean and simple user interface to support novice users unfamiliar with this area. The code of Myscéal is open-source and available on GitHub⁸.

In addition, we have also shown how Myscéal updated both the back-end engine and the front-end interface through the competitions in which the system participated. To join the current trend of using the embedding technique in lifelog retrieval, we have also proposed an embedding-based approach for Myscéal to participate in LSC'22 and LSC'23. Amongst the four LSCs, our Myscéal achieved first place in three of them: LSC'20, LSC'21, and LSC'22. In addition, Myscéal also participated in ImageCLEFlifelog'20, obtaining a considerable third place in ImageCLEFlifelog'20. The competitive result of Myscéal in these competitions has shown the effectiveness of the system. Therefore, I consider the Research Question 1, *How to design a state-of-the-art interactive lifelog retrieval system that assists a novice user to quickly locate items of interest from a conventional multimodal lifelog?*, to be answered by Myscéal, which is a good baseline for future lifelog retrieval systems to be compared with, proven in three years in a row against different systems. Hence, Myscéal will be compared to in future chapters of this thesis.

⁸<https://github.com/allie-tran/lsc-backend/>,
[lsc-processing](https://github.com/allie-tran/lsc-processing), and <https://github.com/allie-tran/lsc-UI>

<https://github.com/allie-tran/>

Chapter 5

Contextual Lifelog Question Answering

In this chapter, Research Question 2 is addressed, which is **How can we evaluate different approaches to question answering on lifelog datasets?** The aim of this research question is to evaluate the application of QA techniques to lifelogs and improve their ability to interpret the meaning and context behind the data, e.g, to reason about the interrelationships of the objects in an image. As lifelog QA is a new task, there is no existing dataset for this purpose. Therefore, the first step is to address the challenges in constructing a lifelog QA dataset by defining the task and the dataset requirements. In Section 5.2, I will describe the process of building the first lifelog QA dataset and present the dataset analysis. A pilot experiment was conducted to determine the baseline models for the lifelog QA task which is described in Section 5.4. The results of the pilot experiment are used to benchmark the lifelog QA dataset with more recent state-of-the-art models in the field of video QA. Finally, I will conclude this chapter with a discussion of the results and future work in Section 5.6.

5.1 Task Definition and Dataset Requirements

Before constructing the dataset, I first define the task of lifelog QA and the requirements for the dataset. First of all, **‘Lifelog Question Answering’ (lifelog QA)** can be viewed as the task of producing a correct answer to a given textual representation of an individual’s information need concerning a past moment or experience from a lifelogger’s daily life. There is *no restriction on the scope* of the questions, which can be compared to the open-domain QA task. On the other hand, the focus of this chapter is on a smaller scope of lifelog QA, which is denoted as **‘Contextual Lifelog Question Answering’** within this dissertation. This distinction is made to distinguish it from the broader open-domain lifelog QA task, although the term ‘lifelog QA’ might be used interchangeably in existing

literature. Contextual lifelog QA involves answering questions about a lifelog *within a provided context*, wherein the context is a lifelog segment that the question refers to. This segment can be a single time point or a time interval with all the associated data such as point-of-view images, location, timestamp, and other sensor data. The questions are presented in natural language a natural language question that can be answered by the lifelog segment.

My objectives are aligned with the following requirements for the lifelog QA dataset:

- **Real-world data and rich metadata:** In order to gain an understanding of what questions are often asked about lifelog data and inform future research directions, the dataset should be based on real-world lifelog data. Rich metadata can also allow more complex questions to be asked and, thus should be included in the dataset. For these reasons, I decided to extend upon the LSC’20 collection [68] as the basis for the dataset.
- **Natural language:** The dataset should contain natural language questions and answers. In addition, I would like to avoid the classification approach where the answers are selected from a list of predefined answers. However, as with any new task, it is important to start with a simple approach and gradually increase the complexity. Therefore, including yes/no questions and multiple-choice questions in the dataset is a good starting point.
- **Open-source:** The dataset should be open-source to encourage more researchers to participate in and explore this research area further. However, only the question and answer pairs are published, not the lifelog data itself, though it is possible to obtain the data through the LSC process.
- **Resource-efficient construction:** As this research is the first attempt at lifelog QA, I will start with a small dataset and gradually increase the size. Moreover, the dataset should be constructed in a resource-efficient manner to create a balance between comprehensiveness and resource conservation. Thus, a decision was made to develop an automated system to generate questions and answers from the lifelog descriptions, which are collected from volunteers.
- **No Unanswerable Questions:** The dataset should not contain unanswerable questions, such as questions proposed in the SQuAD 2.0 dataset [169]. Although the inclusion of unanswerable questions can be useful for the task of open-domain QA, at this initial stage, it is important to ensure that the questions are answerable to avoid confusion and to provide a clear benchmark for the task. However, this is a potential future research direction.

5.2 Dataset Construction

In this section, a detailed explanation of how to build the first Contextual lifelog QA dataset is covered, addressing research question 2.1: **How to adapt existing lifelog test collections to evaluate approaches to lifelog question answering?**. This process is part of my contribution to the field of lifelog QA. To save time and effort, automated steps were applied where possible. The pipeline of the entire three-part process is summarised in Figure 5.1 and the description of each component is as follows:

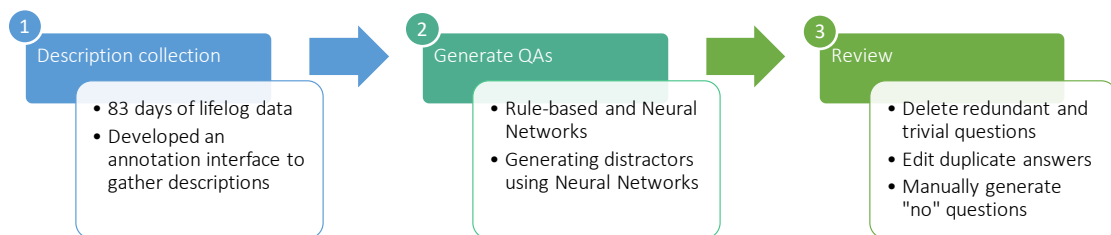


Figure 5.1: The process of dataset construction.

5.2.1 Description Collection

The Contextual lifelog QA data for this work was based on the LSC’20 collection [68]. In total, 26 days of data from the year 2015 and 59 days in 2016 were completed. Each day was segmented into short events of the date based on the locations and activities of the lifelogger to encourage the annotators to focus on individual events. From the provided metadata throughout the day, whenever there was a change in the location (work, home, etc.) or activity (walking, driving, etc.), a new segment would be created. The process resulted in a total of 2,412 segments for annotating. More details about the number of days and images in the dataset can be found in Table 5.1.

Annotators, who were volunteers from undergraduate Computer Science programmes, were asked to describe the events happening in each segment as seen in Figure 5.2. This annotation system was developed to present annotators with all images in each segment along with the metadata such as time, GPS location, and the relative position of the segment in the whole day. Every annotation was accompanied by its starting and ending times. The descriptions include actions or activities; objects that the lifelogger interacted with along with their properties such as size, shape, or colour; the location where the lifelogger was in, heading towards to or away from; and people (with a general identity description to preserve privacy). One example could be ‘The lifelogger is reading a book in a cafe with a person in a black t-shirt.’

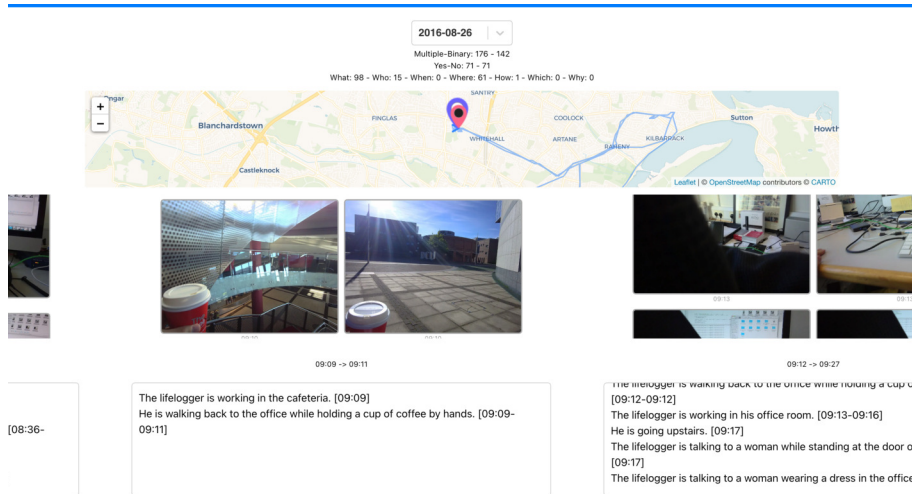


Figure 5.2: Annotation Interface

5.2.2 Generate Question and Answers

The descriptions were converted to a list of questions by an automatic system which is summarised in Figure 5.3. Entity extraction and syntax transformation (ST) were done using hand-crafted rules based on POS tags and Semantic Role labels. To generate question words (who, what, where, etc.), a Seq2Seq neural network was trained on the questions and answers in CoQA [171] dataset. False answers, aka distractors, are generated using RACE [57] with the gathered knowledge from ConceptNet [198] facts as context.



Figure 5.3: The procedure of question-answer generation.

Given the description ‘The lifelogger was reading a book in a cafe’, the generation process can be as follows:

Entities extraction *The lifelogger, reading a book, and in a cafe* are examples of entities in the sentence. I will choose *reading a book* in this example to illustrate further. Thus, the correct answer to this generated question-answer pair would be *reading a book*;

Syntax Transformation — yes/no By moving *was* to the beginning of the sentence, we get ‘Was the lifelogger reading a book in the cafe?’ — ‘Yes’ as a yes/no question-answer pair;

Syntax Transformation — multiple First, based on the POS tags, an automated process

decides the entity is a *phrasal verb*, thus by replacing it with *doing* in the sentence and by applying a rule-based syntax transformation, we get ‘[...] was the lifelogger doing in the cafe?’

Wh-word generation Since questions in this dataset start with a Wh word, a pretrained UniLMv2 model [17] chooses the appropriate question word for this question. In this case, a sensible one would be *What*.

Distractors generation So far, we get the question-answer pair as ‘What was the lifelogger doing in the cafe?’ — ‘Reading a book’. To make this a multiple-choice question, I used RACE [57], a distractor generator for reading comprehension questions, and get the other wrong answers as ‘Using his phone’, ‘Drinking coffee’, and ‘Playing football’.

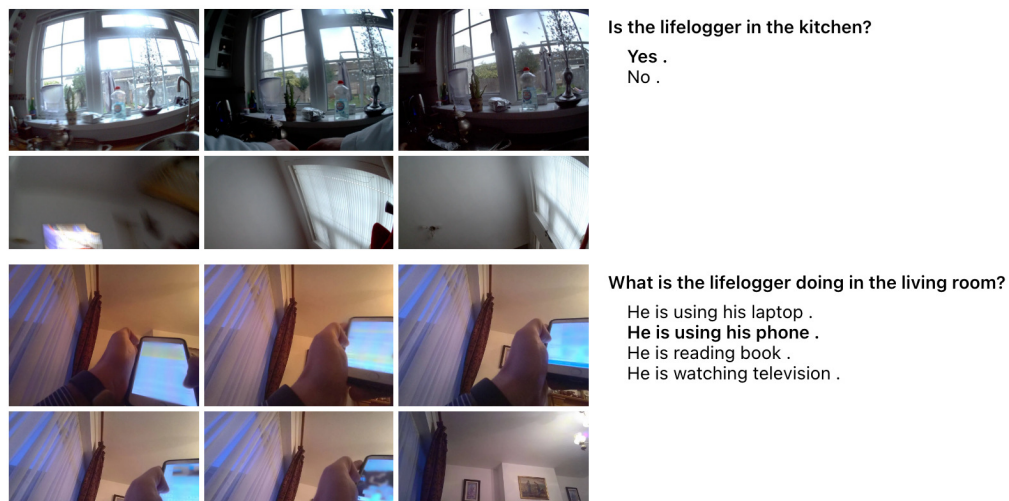


Figure 5.4: Two example question-answer pairs in the dataset. The dataset contains both yes/no questions and multiple-choice questions.

5.2.3 Review

The generated questions and answers are reviewed by the annotators to correct semantics and delete duplicates, as well as to ensure the following constraints:

1. There are no duplicate answers for the same question;
2. The ratios between yes and no questions are balanced. As the automatic syntax transformation could only generate positive yes/no questions, the annotators are asked to create negative ones manually.

5.3 Dataset Analysis

This section presents the analysis of the Contextual lifelog QA dataset and instructions on how to access and use it. Moreover, the limitations of the dataset are discussed, and potential future research directions are proposed.

Table 5.1: Numbers of questions in each month in LSC’20 lifelog data collection.

Month	#Days	Days	#Images	#Questions
Feb, 2015	06	Feb 24–28	8549	941
Mar, 2015	20	Mar 01–20	28563	2745
Aug, 2016	24	Aug 08–31	32026	4871
Sep, 2016	30	Sep 01–30	51195	5595
Oct, 2016	05	Oct 01–05	7375	913
Total	85	—	127708	15065

This new dataset contains 15,065 QA pairs in total. Examples of the QA pairs can be seen in Figure 5.4. On average, the questions contain 7.66 words. Correct answers tend to contain 3.57 words compared to 4.34 words in the generated wrong answers. Table 5.1 presents the breakdown of questions generated. On average, each day contains approximately 177 questions, generated from 1,500 lifelog images. The most questions were generated from September 2016, which is the month with the most days and images. Various question types are included in the dataset, as seen in Figure 5.5. As expected, the majority of the questions are *what* questions, followed by *where* and *who* questions. Although time is a crucial aspect of lifelog retrieval, since the context of the question is provided, *when* questions are not as common.

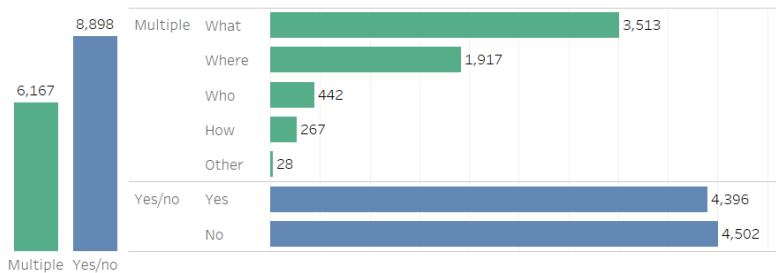


Figure 5.5: Numbers of each question type in Contextual lifelog QA dataset.

Figure 5.6 shows the distribution of the first four words in the questions. Interestingly, almost half the multiple-choice questions are ‘What is the lifelogger doing?’, which is a vague question that can be answered by many different actions. The second most

not as good as the human-generated questions. For example, general questions such as ‘What is the lifelogger doing?’ are generated more often than specific questions such as ‘What is the lifelogger doing in the office?’. Other issues include the lack of diversity in the generated questions, which are often repetitive and not entirely relevant to the context of the lifelog data. This is made worse by the fact that the lifelog collection is based on a single lifelogger. However, I believe that the dataset is still useful for the task of lifelog QA as it is based on real-world lifelog data and is the only such dataset available that has been created for this purpose.

Another important factor is the level of inter-annotator agreement. During the review process, the annotators were asked to review other annotators’ questions and answers to avoid bias. However, no formal inter-annotator agreement was conducted. Due to the large number of questions, it was a challenge to address this issue under the time and resource constraints. This is a potential future research direction.

5.4 Evaluation

In this section, I present benchmark experiments to evaluate different approaches to the contextual lifelog QA task on the dataset. These experiments aimed to determine the baseline models for the task. The results of these experiments are used to answer Research Question 2.2, which is, **What existing question answering techniques are most effective when applied to lifelog data?**

As the dataset consists of yes/no questions and multiple-choice questions, accuracy was used as the evaluation metric, which is the proportion of correct answers to the total number of questions. This straightforward metric is suitable for the task of Contextual lifelog QA as the questions are not open-ended and the answers are limited to a small set of options.

Accuracy is the normalized criteria for assessing the quality of the generated answer based on the testing question set. It is given by,

$$Accuracy = \frac{1}{Q_t} \sum_{q \in Q_t} \left(1 - \prod_{i=1}^M 1[a_i \neq o_i] \right)$$

where Q_t is the question set, the generated answer words $Oq = (o_1, o_2, \dots, o_M)$, and the ground-truth words $Aq = (a_1, a_2, \dots, a_M)$. Accuracy = 1 indicates that the generated text is the same as ground truth and accuracy = 0 indicates that the generated text is different from the ground truth. The accuracy is calculated for each question and then averaged over the entire question set.

5.4.1 Baselines

The baselines for the Contextual lifelog QA task were determined by conducting a pilot experiment. The aim of this experiment was to determine the targeted performance of the dataset and to evaluate the application of existing QA models to the task.

Human Gold-standard

To determine the targeted performance (in terms of accuracy) on the dataset, I performed a user study, asking different groups of 10 volunteer students to complete the question-answering task. Each volunteer was asked to answer 20 yes/no questions and 20 multiple-choice questions chosen randomly from the testing set. Each question was accompanied by the relevant images. To avoid bias, there was no overlap between the annotators that have worked on the questions and the students participating in this study. The gold standard accuracy was found to be 84.17% for yes/no questions and 0.8625 for multiple-choice questions. The reason that the scores are less than 1.0 is because the volunteers were presented with the relevant section for the question, rather than the lifelog data for the whole day, so in some cases, they did not fully understand the context of the event mentioned in the question. Another interesting feedback from the participants, as well as the annotators, concerns the volume of lifelog data causing issues in understanding. This is a common problem in lifelog analytics when the decisions regarding lifelog data are often made by a third party and not the original data-gathering lifelogger, for example, as seen in the studies carried out by Byrne et al. [25].

Question-only

Several heuristic baselines were implemented, which use only the questions and their candidate answers in a similar approach to Castro et al. [26]. Specifically, *Longest answer* and *Shortest answer* choose one out of the four options with the most or the fewest number of tokens, respectively. *Word matching* chooses the answer based on the number of tokens they have in common with the question. Because yes/no answers have no difference either in length or the number of common words with the questions, these models were omitted for this experiment.

Moreover, I also experimented with a *Sequence-to-sequence (S2S)* model based on the architecture of UniLMv2 [17], the state-of-the-art model in natural language understanding and generation tasks. S2S was trained on the CoQA [171] question-answer pairs. It encodes the question with a 2-layer LSTM, then encodes the candidate answers and assigns a score to each one. The text is tokenised and represented using Glove 300-D embeddings [163].

Question and Vision

Because of the similarity to Video QA task, I applied a video QA model called *TVQA* [116], trained on the TVQA dataset described in the same paper. TVQA is a multimodal model that uses Faster R-CNN [173] and ResNet101 [73] to extract visual features from the video frames, and Glove 300-D embeddings [163] to encode the question, the subtitles, and the answers. Bi-directional LSTMs were used to encode textual and visual sequences, and a context-matching module was used to incorporate subtitles into the video features. The model was trained on the TVQA dataset, which consists of 152,545 QA pairs from 21,793 video clips from 6 TV shows. This was the state-of-the-art system in Video QA during the time of writing.

To evaluate the application to lifelog data, I considered each day to be a one fps video with each image (along with the attached metadata) as one single frame in that video. I converted the annotated starting and ending times into the ordinal index of the frames in the video. Moreover, the subtitles, intended for videos, were replaced with a concatenation of metadata associated with the frames. While it may seem strange to treat visual lifelog data as motion video, it is temporal in nature, and many of the participants in the LSC challenge [68] have modified existing Video Search systems from the VBS challenge [130] to treat lifelog data as 1fps video.

Results

Both S2S and TVQA models were retrained on the training set of the LLQA dataset and achieved a small improvement in accuracy compared to the untrained versions, as seen in Table 5.2. Furthermore, there is no considerable difference between the question-only models. Although the average length of the correct answers is shorter than the wrong ones, *Shortest answer* did not perform well at the lowest accuracy of 17.17% for multiple-choice questions. Among the models, the retrained TVQA achieved the best performance with an accuracy of 63.38% and 61.36% for yes/no questions and multiple-choice questions, respectively. However, humans still significantly outperformed the models. The results highlighted that the existing approaches are still far from the human gold standard for the Contextual lifelog QA task, so they should be optimised to improve performance. This will be a potential and promising topic for future research in lifelog domain in general, and especially in lifelog QA.

5.4.2 Pretrained Video-Language Models

Based on the literature review conducted in Chapter 2 and the results of the pilot experiment, I benchmarked the LLQA dataset with more recent state-of-the-art models based on the following criteria:

Table 5.2: Accuracy (%) of different models in the pilot experiment.

Model	Yes/no	Multiple-choice
S2S	52.06	31.48
S2S (retrained)	50.66	36.26
TVQA	49.56	40.85
TVQA (retrained)	63.38	61.36
Gold standard	84.17	86.25

- **Multimodality:** The model should be able to process both visual and textual inputs. Lifelog data is multimodal in nature, consisting of images and textual meta-data. Moreover, the questions and answers themselves are presented in natural language form. Therefore, I specifically focus on models that can integrate and process both modalities. Video QA models are a particularly good fit for this task as they are designed to process both visual and textual inputs.
- **Pretraining:** Models pretrained on large-scale datasets are preferred as they have been shown to achieve state-of-the-art performance on various downstream tasks. The more comprehensive the pretraining dataset is, the better the model can generalise to other domains. WebVid2.5M and WebVid10M [14] are large-scale multimodal datasets that contain millions of video clips with billions of text-video pairs. Specifically, WebVid10M is the largest video-language dataset to date and has been used to pretrain many state-of-the-art models.
- **Performance on Video QA:** The model should have been shown to achieve state-of-the-art performance on video QA tasks. This is to ensure that the model is capable of processing the multimedia nature of lifelog data.
- **Fine-Tuning flexibility:** The model should be flexible enough to be fine-tuned on the LLQA dataset. This is to ensure that the model can be adapted to the Contextual lifelog QA task.
- **Availability of pretrained weights:** The pretrained weights of the model should be available to the public and compatible with the chosen deep learning framework (PyTorch). Accessible pretrained weights allow model integration and experimentation.
- **Support for multiple-choice questions:** Since the task of visual QA and video QA is oftentimes formulated as a classification task, not all models support multiple-choice questions. Handling a diversity of question types is important for the broader

lifelog QA task as it is open-domain and the questions can be of any type.

- **Hardware suitability:** Inference and finetuning of the models should be possible on the available hardware. This is to ensure that the models can be run effectively.

By considering the above criteria, I selected the following models for evaluation: Singularity [115], FrozenBiLM [229], and VioletV2 [53]. All these models share some common characteristics: they are pretrained on large-scale vision-language datasets, including both images and videos; they include a text encoder, a vision encoder, and a cross-modal module. The text and vision encoders are initialised by using pretrained weights from popular models, such as BERT [38], RoBERTa [127], and UniLM [45] for language, and ResNet [73], Swin Transformer [128], and CLIP [167] for vision. The main difference between the models is in the cross-modal module, which is used to learn the relationship between the text features and the visual features. Furthermore, pre-training objectives and training strategies also play an important role in the performance of the models. The details of each model are as follows:

- **Singularity**[115] adopts a single-frame training approach with random sampling and multi-frame inference for an efficient and accurate learning process. The model was pre-trained on a set of video-text tasks with three pre-training objectives: (1) contrastive loss on video-caption pairs, (2) masked language modelling (MLM)[38] to predict masked tokens from video captions, and (3) vision-text matching (VTM) to predict the matching score between video and text. Several datasets were used for pretraining, including COCO [126], Visual Genome [104], SBU Captions [161], CC3M [192], CC12M [28], and most importantly WebVid [14].

Singularity showed competitive performance on various downstream video-text tasks such as text-to-image retrieval, image question answering, text-to-video retrieval, and video question answering. In this work, I chose the model that was finetuned for the video QA task with MSRVTT-QA dataset [224], which contains 244K open-ended questions on 10K MSRVTT videos. The model was then further finetuned on the LLQA dataset for 10 epochs using AdamW [135] with an initial learning rate of $1e-4$. Two settings were considered: (1) Singularity (1-frame) where the model was trained on a single frame, and (2) Singularity-temporal (4-frame) where the model was trained on four randomly sampled frames. At inference time, the model was evaluated on four uniformly sampled frames.

- **FrozenBiLM**[229] uses a transformer-based cross-modal encoder to connect the two modalities. The cross-modal transformer is also initialised with pretrained weights from an MLM such as BERT [38]. However, lightweight adapter modules are added between the cross-modal layers to adapt the pretrained weights to the downstream task. As per the name of the model, all encoders except the adapters are frozen during training. Thus, the model is able to achieve good performance with a small number of training examples. The pre-training objective of FrozenBiLM is similar to the MLM objective, where the model

is trained to predict randomly masked tokens based on the surrounding text tokens and the video input. The model, pre-trained on WebVid [14], is able to perform zero-shot video QA through MLM, where the answer is predicted by filling in the mask token in a prompt sentence. Its prompt approach gives the model the ability to solve various video QA formats such as yes-no, multiple-choice, open-ended, and fill-in-the-blank questions.

ActivityNet-QA [236] is a large-scale video QA dataset that contains 58,000 QA pairs on 5,800 complex web videos. To minimise the domain gap between the pretraining and the LLQA dataset, FrozenBiLM was finetuned on ActivityNet-QA for 20 epochs by the original authors, and then further finetuned on the LLQA dataset for 10 epochs. In order to take lifelog metadata into account, a prompt such as ‘The event happened at [time] in [location]’ is constructed to pass to the model in place of video subtitles.

- **VioletV2**[53] is a fully end-to-end VIdEO-LanguagE Transformer model, consists of a Video Swin Transformer (VST)[158] to compute temporal-aware features from video input. The VST was initialised with weights from a pretrained model on Kinetics-600 dataset [134]. The language encoder and cross-modal transformer are initialised from the pretrained BERT-base model. Other than the VTM and MLM pretraining objectives similarly to Singularity, VioletV2 also uses Masked Visual Modelling (MVM) objective to reconstruct the visual tokens from the video input. The model was pre-trained on WebVid2.5 [14] dataset, fine-tuned on MSRVTT-QA [224] dataset, and further fine-tuned on the LLQA dataset for 10 epochs.

Dedicated Models for LLQA

To provide a more comprehensive comparative analysis, I produced and evaluated a set of dedicated models for the Contextual lifelog QA task. They are necessary due to the unique demands and challenges posed by lifelog question answering. Existing models may not be optimised for this specific task, which involves understanding long-term, multimodal data and answering questions based on it. For example, the sampling rate of lifelog data is much lower than that of videos, which can be up to 30 frames per second. Meanwhile, most pretrained models only sample a small number of frames (for example, 4 frames in Singularity) from the video in order to reduce the computational cost of pretraining. This may not be suitable for lifelog data as the sampling rate is too high. Furthermore, the flexibility to experiment with different architectural choices is important for the lifelog QA task as it is still in its infancy.

Following a design similar to pretrained models above, the dedicated lifelog question answering models should leverage the power of pretrained embedding models, such as CLIP [167], to generate embeddings for images, questions, and candidate answers. CLIP embeddings offer an effective means of encoding both textual and visual information into a shared embedding space, providing a rich source of knowledge and context.

In this approach, for each LLQA question, a question embedding Q is constructed by

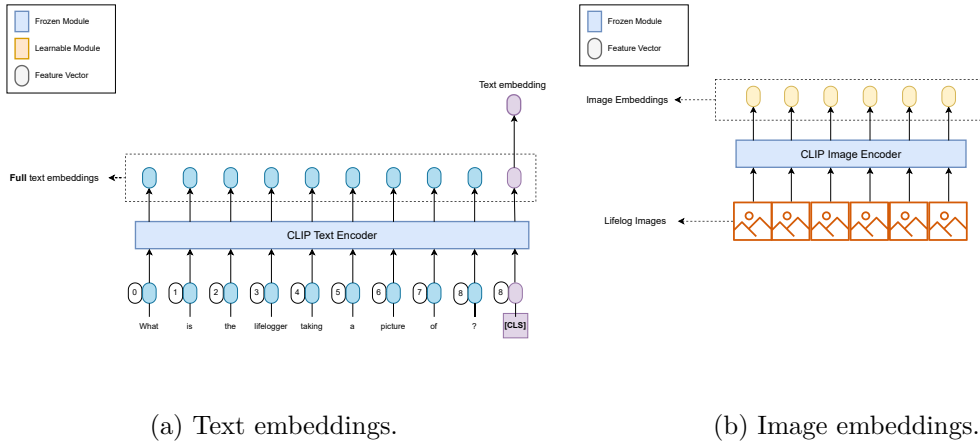


Figure 5.7: Getting the CLIP embeddings. The weights of the CLIP model are frozen (shown in blue) during training.

feeding the question into the CLIP text encoder as seen in Figure 5.7. Most of the time, the [CLS] token embedding is used as the question embedding. However, I also experimented with using the full sequence of the last hidden states to represent the question in one of these models (FullCrossQA). This is to ensure that the model can take advantage of the full question information. For the candidate answers, I concatenate the question with each of the answer choices to form individual prompts, such as ‘*Question: What is the lifelogger doing? Answer: Reading a book.*’. These prompts are also encoded by the CLIP text encoder to generate answer embeddings A_1, A_2, A_3, A_4 . At the same time, relevant images to the LLQA question are extracted and encoded by the CLIP image encoder to generate image embeddings I_1, I_2, \dots, I_N . The CLIP model is frozen during training, similar to FrozenBiLM. This is partly due to the limited computing resources available, and partly to ensure that the model is not overfitted to the LLQA dataset and retains the powerful knowledge learned from pretraining.

In order to take metadata into account, following FrozenBiLM’s approach of using subtitle prompt, I construct a textual description from the accompanying metadata of the relevant images in the form of ‘*The event happened at [time] in [location]*’. CLIP text encoders are then used to encode this description into a metadata embedding M .

The primary objective of these models is to calculate a logit score for each candidate answer. Cross entropy loss is employed to train the models based on the logit scores and the ground truth labels. To produce the logit scores, cosine similarity is calculated between the global image embedding G and the answer embeddings A_1, A_2, A_3, A_4 . The details of each model on how the global image embedding and the answer embeddings are generated are as follows:

- **MeanQA** is a simple baseline model for this experiment in which the image embeddings are processed by a mean pooling layer to produce a single global embedding G .

To produce a metadata-aware global embedding G_m , the metadata embedding M can be concatenated with the image embeddings before feeding them into the mean pooling layer. The question embedding is ignored as it already appears in the candidate answer prompts. The model is shown in Figure 5.8. The metadata embedding can be concatenated with the image embeddings to produce a metadata-aware global embedding G .

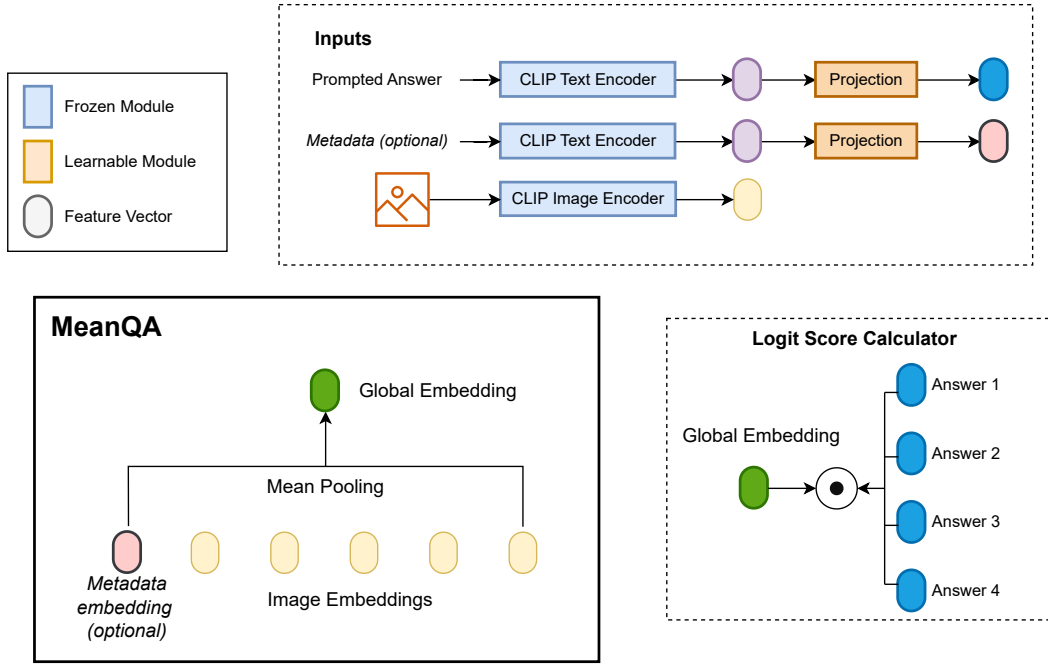


Figure 5.8: MeanQA model.

- **SelfQA** enhances MeanQA by employing a transformer layer to learn the temporal dependencies between the images, as seen in Figure 5.9. Transformers are renowned for their ability to capture long-term dependencies in sequential data, which is a key feature of lifelog data. Positional embeddings are added to the embeddings to provide positional information before feeding them into the transformer layer. Average pooling is then applied to the output of the transformer layer to produce a global embedding G .

To produce a metadata-aware global embedding G_m , I concatenate the metadata embedding M with the image embeddings before feeding them into the transformer layer. However, the metadata token is not included in the last average pooling layer since its information has already been incorporated into the other hidden states due to the self-attention mechanism. The rest of the model is the same as MeanQA.

- **CrossQA** is a more complex model that employs a cross-modal transformer layer to learn the cross-modal interactions between the images and the question. The ques-

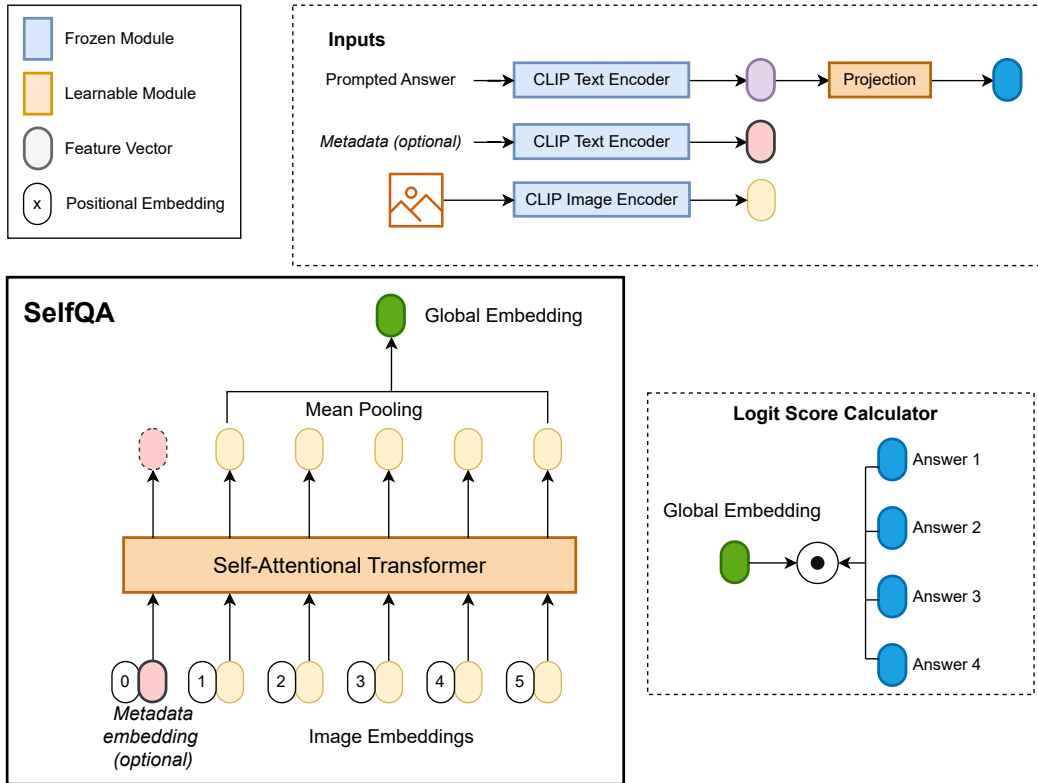


Figure 5.9: SelfQA model.

tion embedding Q is concatenated with the image embeddings before feeding them into the cross-modal transformer layer. Positional embeddings are added in the same way as SelfQA. The output of the cross-modal transformer layer is then processed by a mean pooling layer (excluding the one corresponding to the question embedding) to produce a global question-guided embedding G . Regarding metadata information, the metadata embedding M can be inserted after the question embedding and before the image embeddings as seen in Figure 5.10.

Instead of using the answer embedding A_1, A_2, A_3, A_4 from the CLIP model, the first output of the cross-modal transformer layer, as shown in the blue oval in Figure 5.10, is used for the logit score calculation. This output is a cross-modal embedding that incorporates visual information to help answer the question.

- **FullCrossQA** employs fully the last hidden state of the transformer layer in the text encoder (instead of only the [CLS] token embedding) and concatenates it with the image embeddings to produce a question-aware visual embedding. In addition to the positional embeddings, modality embeddings are also integrated to indicate the type of input (text or image). The answer embeddings are generated by taking the output of the cross-modal transformer layer that corresponds to the [CLS] token embedding, as shown in Figure 5.11.

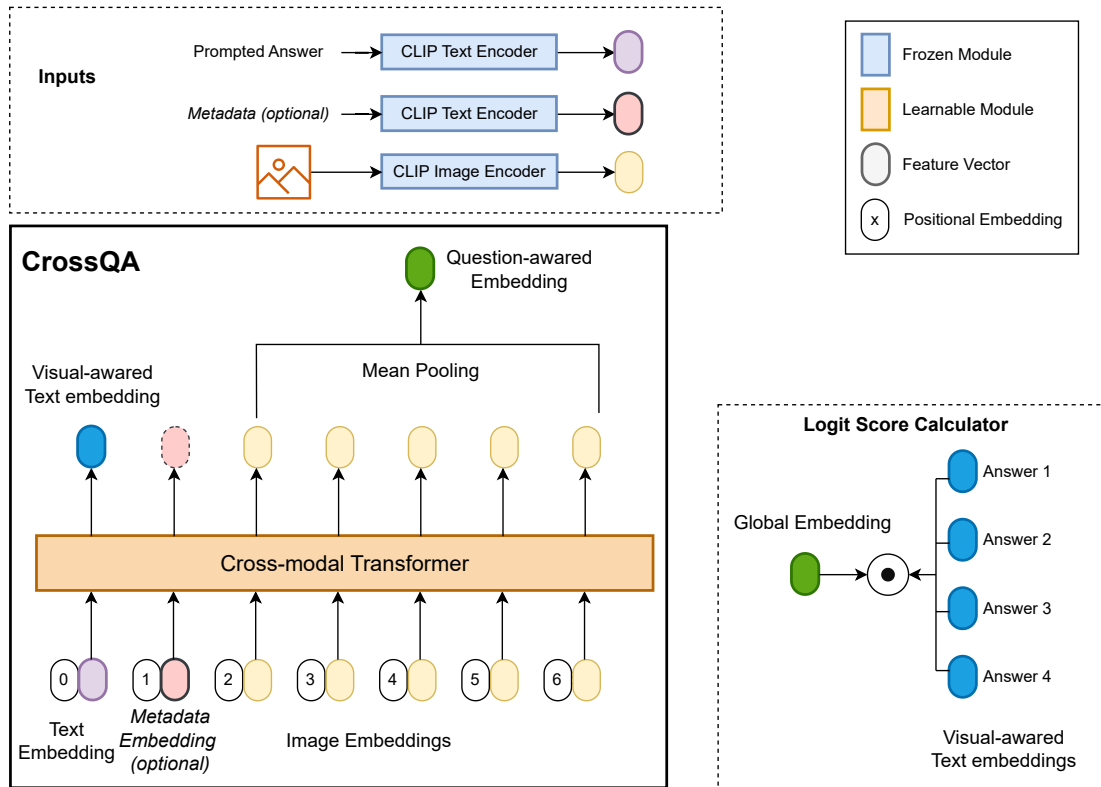


Figure 5.10: CrossQA model.

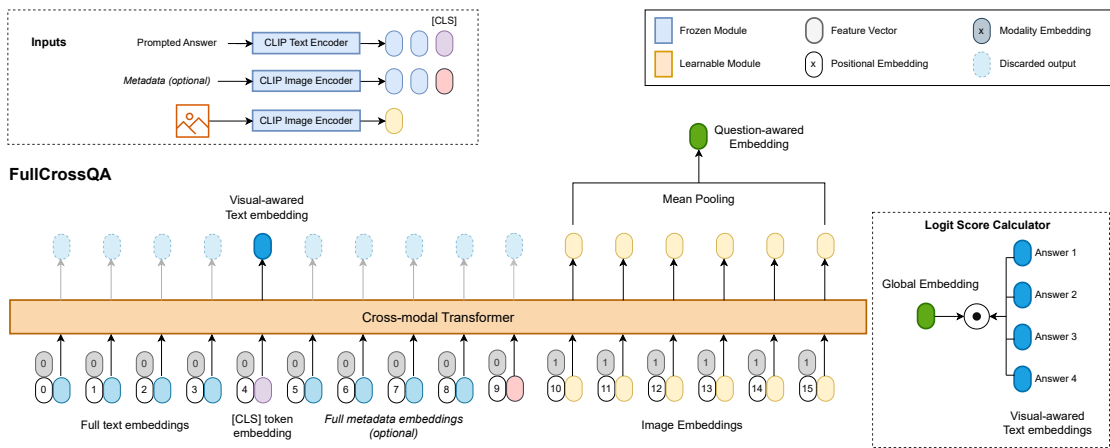


Figure 5.11: FullCrossQA model.

5.4.3 Benchmarking Results

Table 5.3 shows the results of the benchmarking experiments. It is clear that utilising large-scale pretrained text and vision models can provide a significant boost in performance compared to the baseline models discussed in the previous section. Interestingly, for pretrained VideoQA models, the overall performance of yes/no questions is higher than

Table 5.3: Results of more recent SOTA models in video QA on LLQA dataset. All models were evaluated on both yes/no and multiple-choice questions at the same time (Overall). To indicate the use of metadata, (+m) is added to the model name. WV stands for WebVid.

Model	Vision	Text	Multiple-		Overall
	Encoder	Encoder	Yes/No	choice	
Singularity (1-frame)[115]	BEiT [16]	BERT [38]	73.79	61.66	69.16
Singularity (4-frame)	BEiT	BERT	73.2	60.76	68.46
FrozenBiLM [229]	CLIP [167]	DeBERTa [74]	72.87	71.14	72.21
FrozenBiLM (+m)	CLIP	DeBERTa	73.12	71.32	72.43
VioletV2 [53]	VST [158]	BERT	64.67	56.83	61.67
MeanQA	CLIP	CLIP	66.10	68.40	66.98
MeanQA (+m)	CLIP	CLIP	67.94	66.31	67.32
SelfQA	CLIP	CLIP	66.29	71.44	68.25
SelfQA (+m)	CLIP	CLIP	66.65	70.84	68.25
CrossQA	CLIP	CLIP	68.38	71.02	69.39
CrossQA (+m)	CLIP	CLIP	67.69	72.69	69.66
FullCrossQA	CLIP	CLIP	70.44	72.51	71.23
FullCrossQA (+m)	CLIP	CLIP	69.15	72.51	70.43

that of multiple-choice questions. The opposite is true for the dedicated models. This is likely due to the specific prompt-based approach used by these models where the two candidate answers are only different in one word (‘yes’ or ‘no’) while containing the same context thus providing less information for the model to distinguish between the two. Singularity (1-frame) achieves the best result for yes/no questions with an accuracy of 73.79%. Meanwhile, FullCrossQA achieves the best result for multiple-choice questions with an accuracy of 72.93%. This is likely due to the fact that the tailored models are specifically designed and trained for the LLQA dataset, therefore they are more suitable for this task. Overall, FrozenBiLM achieves the best performance with an accuracy of 72.43% over both question types. Thus, I believe that FrozenBiLM is the most suitable model for the task of lifelog QA in its current state.

Regarding incorporating metadata as part of the input, it provides an insignificant improvement in performance in most models except for FullCrossQA. This is likely because most of the questions in the dataset are based on the images, rather than the metadata. In the specific case of FullCrossQA, feeding the full length of the metadata into the transformer layer might have caused the model to overfit to the metadata (since the length of text sequences in CLIP’s text encoder is set at 77 tokens, which is much longer than the average number of images in each question). Therefore, the model is not able to learn the

relationship between the images and the metadata effectively. Despite this, I believe that the metadata is a crucial part of the lifelog QA task as it provides additional information about the lifelogger’s actions. Therefore, I would like to encourage more researchers to explore this area further.

One interesting observation is that the custom-built models have similar performance to each other, despite their different architectures. This suggests that the transformer layer is not able to learn the temporal dependencies between the images effectively. In this experiment, only one layer of transformer as experimenting with a higher number of layers did not improve the performance while significantly increasing the training time. This could be due to the fact that the LLQA dataset is relatively small. Furthermore, the sampling rate of lifelog data could be too low for the transformer layer to detect the flow of events. For these reasons, I believe this is an important area for future research.

5.5 Discussion

In this section, I discuss the findings of the experiments and provide insights into the performance of the models. I also highlight the strengths and limitations of each model and propose potential research directions for future works. This section aims to answer Research Question 2.2, **What existing question answering techniques are most effective when applied to lifelog data?**

In summary, the pretrained models achieved better accuracy than the baseline models. The tailored models for LLQA achieved comparable performance to the pretrained models, suggesting that they are also suitable for the task of Contextual lifelog QA. However, given the flexibility and robustness of lifelog QA due to their ability to generate answers outside the candidate answer set. This is a significant advantage over the dedicated models, which can only choose from the candidate answers. Furthermore, the pretrained models are more equipped to handle out-of-domain data, which is important as lifelog data is highly diverse and can be collected in various environments. Therefore, at the moment, FrozenBiLM is the most suitable model for the task of lifelog QA and is chosen as the baseline model for the rest of the experiments in this thesis.

The template of using pretrained text and vision models to generate embeddings for the images, questions, and candidate answers is a promising approach for the lifelog QA task. It is flexible and can be adapted to different architectures. Furthermore, pretraining on a large-scale dataset such as WebVid10M [14] can provide a significant boost in performance. Masked language modelling (MLM) is a suitable pretraining objective for lifelog QA as it can be used to generate answers outside of the candidate answer set, as well as to generate free-form answers. Furthermore, the pretrained models can be fine-tuned on the LLQA dataset to improve their performance. However, this is not possible in this thesis due to the limited resources in storage and computing power.

Some video QA models incorporate motion features such as C3D [203], whose equivalent for lifelog data is not available. It is unclear whether the transformer layer is able to learn the temporal dependencies between the images effectively given the insignificant differences between the average pooling approach and the transformer approaches described in the previous section. Therefore, I believe that the development of a motion feature extractor for lifelog data is a promising research direction. Moreover, the sampling rate of lifelog data is much lower than that of videos, which can be up to 30 frames per second. Meanwhile, most pretrained models only sample a small number of frames (for example, 4 frames in Singularity) from the video in order to reduce the computational cost of pre-training. It is also because most video QA datasets contain short videos. One possible approach is to sample video frames with a lower rate similar to lifelog data and pretrain models based on this. Furthermore, a dedicated structure for other modalities such as metadata can be developed to improve the prompting mechanism.

5.6 Conclusion

In this chapter, I addressed Research Question 2: **How can we evaluate different approaches to question answering on lifelog datasets?**. Specifically, I developed the first Contextual lifelog QA dataset based on the LSC'20 collection. The dataset consists of 15,065 question-answer pairs, which are generated from the descriptions of the lifelog data. The findings suggest that a large proportion of the dataset involves the lifelogger's actions or interactions with other objects, therefore it is crucial to improve the standard action recognition mechanism. Through several baseline experiments, I assessed the suitability of the dataset for the task of lifelog QA. It is noteworthy that there is still a significant gap between the proposed baselines and human performance on QA accuracy, meaning that there is a significant research challenge to be addressed. The dataset is published at <https://github.com/allie-tran/LLQA>. I also included the annotated description with timestamps, which can be used to develop models for lifelog captioning tasks. By creating this dataset, I hope it can encourage more researchers to participate in and explore this research area further.

The second part of this chapter focused on benchmarking the dataset with more recent state-of-the-art models. It was found that the pretrained models achieved significant performance improvements compared to the baseline models. Tailored models performed similarly to pretrained models, indicating their suitability for lifelog QA. However, pretrained video-language models are more appropriate for open-domain lifelog QA due to their flexibility and robustness, which allows them to generate answers beyond the candidate answer set. FrozenBiLM is the most suitable model for the task of lifelog QA in its current state. I believe that the findings of this chapter can provide a solid foundation for future research in lifelog QA. In the next chapters, I will explore the steps of adapting

FrozenBiLM to the lifelog QA task.

Chapter 6

Event-based Embeddings

This chapter is dedicated to addressing research question 3.1, which is *Does the event-based retrieval support the user to achieve comparative performance to image-based retrieval for lifelog data?*. This step involves the introduction of a novel system named **MyEachtra** (/mai-AK-truh/), which is an enhanced version of the Myscéal framework outlined in Chapter 4. However, what sets MyEachtra apart is its nuanced shift towards a bigger retrieval unit, namely ‘events’, from the previous ‘images’ format. This tailored approach acknowledges the temporal nature of lifelog data, an aspect that differentiates the task of lifelog question answering from visual question answering (VQA) tasks. On top of that, the adoption of an event-based approach aligns with the format of the LLQA dataset, detailed in Chapter 5. In this dataset, the provided context for each question is more likely to be a *sequence of images* (and their associated metadata) over a period of time, as opposed to a *single image*. Another motivation for this approach is that it is more intuitive for users to think in terms of events rather than individual images, and showing events instead of images can reduce the amount of repetitive information displayed to the user, as well as increase the amount of context provided. Based on these observations, I hypothesise that an event-based approach can be an effective way to improve the performance of lifelog retrieval systems, create a more intuitive user experience, and better align with the lifelog QA task.

At first, this notion of an event-based approach may seem related to the key-frame extraction process, which is a common practice in video summarisation and retrieval. Video retrieval systems often rely on selecting frames that capture the essential content, known as key-frames, to represent videos [60, 129, 181, 186]. The event-based approach in MyEachtra is actually the *opposite* of such process. Keyframes in video retrieval are extracted so that we can represent videos through images. In contrast, MyEachtra is designed to use *all* available information (in the form of lifelog images and their associated metadata) to represent events. This approach is more aligned with the nature of the lifelog data, where the information need is not naturally captured in a single image. In a bigger

picture, this event-based approach can be extended to video retrieval systems, given that event-embedding models can capture the temporal nature of videos and provide a more effective way to represent videos.

In this chapter, the proposed system, **MyEachtra** (/mai-AK-truh/), builds on the success of Myscéal (E-Myscéal in particular) and includes modifications which are summarised as follows: (i) an event segmentation process influenced by location metadata; (ii) an event-based approach that exploits pretrained image-text cross-embedding models to develop representative embeddings for events; and (iii) a redesigned user interface showing events and highlighting important images with relevant events. The details of these modifications are described in Section 6.1. An analysis of the system’s performance is also presented in Section 6.2. A user study was also conducted to compare the performance of MyEachtra with the previous image-based Myscéal system. The results are presented in Section 6.3. After that, Section 6.4 outlines the performance of MyEachtra on the known-item search (KIS) and ad-hoc queries in LSC’23 [71], compared to other participating systems. Finally, Section 6.6 concludes the chapter.

6.1 MyEachtra

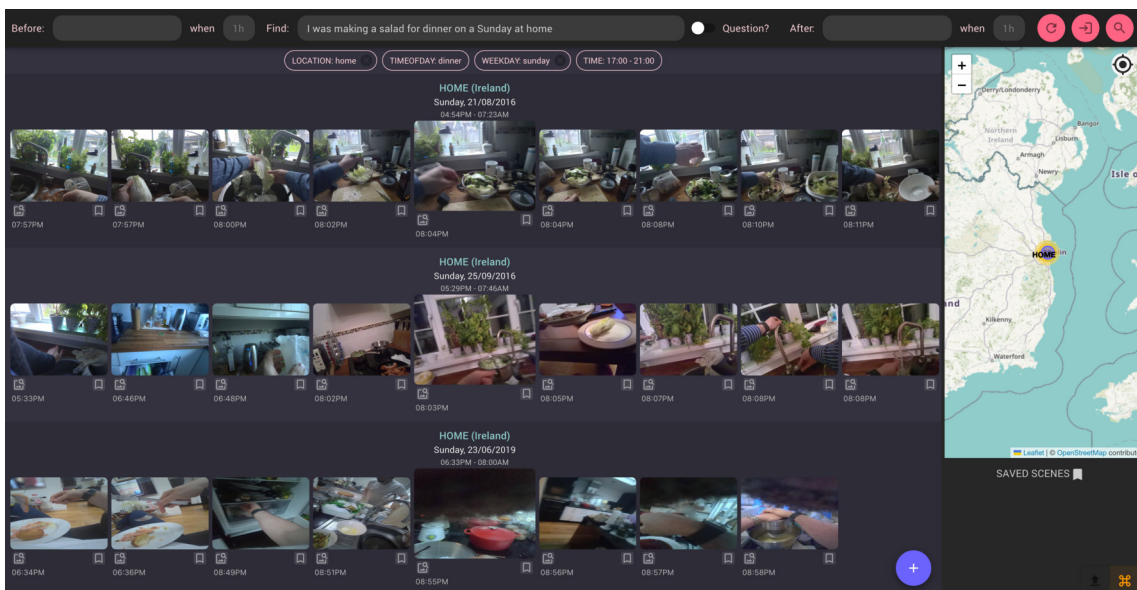


Figure 6.1: MyEachtra’s user interface. Each row represents an event. The most relevant image is highlighted and placed in the middle of the row.

6.1.1 Event-Based Approach

The main enhancement for MyEachtra is that, instead of comparing each image in the dataset to the query using cosine distances, it compares *events*. I will illustrate how to

turn image embeddings into event embeddings. By using CLIP, we denote the pretrained encoders as $\omega(u) = \mathbf{w}$ and $\theta(t) = \mathbf{c}_t$ which encode image u and text t into $\mathbf{w}, \mathbf{c}_t \in \mathbb{R}^d$. Assume an event e is composed of s images such that $e = u_1, u_2, \dots, u_s$. Therefore, the embeddings of images can be joined into a matrix $\mathbf{Z} = [\omega(e) = \mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_s]$, where $\mathbf{Z} \in \mathbb{R}^{d \times s}$. The goal is to find an aggregation function Λ that maps $\mathbf{Z} \in \mathbb{R}^{d \times s}$ into a global event representation $\mathbf{c}_e \in \mathbb{R}^d$. In the context of the LSC, Λ is preferably independent of the query t . Several options are possible from previous work in video-text models.

Mean Pooling A simple yet effective way of combining a list of embeddings is average pooling over the temporal dimension. Mean pooling is often used as a baseline to compare new video models.

Clustering Portillo et al.[164] experimented with clustering the events and selected the cluster centres as representative embeddings. By doing this, one event can be represented by multiple embeddings, addressing different interpretations of the same event. For example, a birthday party can be divided into several smaller activities, such as food preparation, cake cutting, and socialising, all of which can occur concurrently and cannot be segmented in a conventional way. The only change I made from their method was that instead of using K-means clustering method, I employed OPTICS [10], a density-based clustering method, to dynamically address the vastly varied lengths of events, ranging from a single image to a maximum of 297 images.

Transformer encoders The most popular technique for temporal modelling in videos (as well as events in this system) is to use transformer encoders [218] and learn a self-attention mechanism to emphasise important images. Note that since the outputs of transformer encoders are still in a sequential format, they are average pooled to create the global embedding.

Weighted Mean Another way to work with these outputs is passing them through a Linear Layer (where the output dimension is 1) to produce image weights, indicating how important an image is in the event, as described in [15].

After getting the event embeddings, the cosine similarity is still used for the retrieval process. The similarity score is summed with other scores (TF-IDF for location names, GPS filters, time filters, etc.) within ElasticSearch, similar to the E-Myscéal. These are referred to as the *event scores* in the later subsection. In Section 6.2, I will evaluate the results using the previously mentioned options.

6.1.2 Displaying Events

The user interface is redesigned to show the resulting ranked events (rather than images) in a way that is easy to understand and highlights relevant information such as location, time, and highly ranked images within an event. After getting the ranked results from ElasticSearch, to further reduce repetitive information, if there are no location changes between some events, they are merged as one *row* in the user interface.

To find the best images within each *row* to display, first, the cosine similarities between each image and the user’s query are calculated. Each *image-based similarity score* is then multiplied with the *event score* returned by Elasticsearch (as mentioned in the last section). Finally, the softmax function is applied over the scores to get the *image scores* and emphasise the most relevant images and reduce the impact of outliers. This calculation is limited to the top 100 resulting events and repeats when the user requests more results.

Aggregating event images by using Weighted Mean also produces the image weights as the output of the model. In that case, these weights are multiplied by the cosine similarities and event scores before the softmax function is applied.

To help the user quickly identify the most highly scored image within each group, the image with the highest score is highlighted and placed in the middle of the row. This design choice was made to draw the user’s attention to the most important visual information and reduce the need for extra scanning and searching. In addition, up to the next six most relevant images (three on each side, left and right) are also shown to the user as they are not only most likely relevant but also provide additional context to improve the user’s understanding of the event. An example can be seen in Figure 6.1.

In an evaluation or competition setting, the user is asked to submit the most relevant image(s) to the query. For known-item queries, the user can submit an image by clicking on a checked button displayed on it. On the other hand, for ad-hoc queries, users can submit the entire event by holding down the Shift key and clicking on any image’s submit button.

6.2 Evaluation Using LSC’22 Queries

An automatic evaluation was carried out to assess the performance of our event-based approach. All 14 known-item queries of the LSC’22 campaign were used in this pilot study. They are manually split into ‘before’, ‘main’, and ‘after’ hints before requesting the ranked list from the backend system. From the result, we measure the Hit Rate at K ($H@K$), ignoring further actions such as map filtering, temporal browsing, or visual similarity search. Adjustment for events is also applied. In this case, $H@K$ here means that one of the target images is included in the first K event (whose a maximum of 7 images are shown in the user interface). This metric could provide a baseline for the system’s performance and help to compare different approaches.

As mentioned before, in the live LSC event, each query will be gradually revealed to the participants every 30 seconds. These extra hints are expected to provide more information about the content of the correct images and make the search easier. As a result, I evaluated MyEachtra at different steps of the query, from one hint to six hints. Additionally, I experimented with four different approaches to aggregate image features, and the results are shown in Figure 6.2. The configurations of each experiment are as

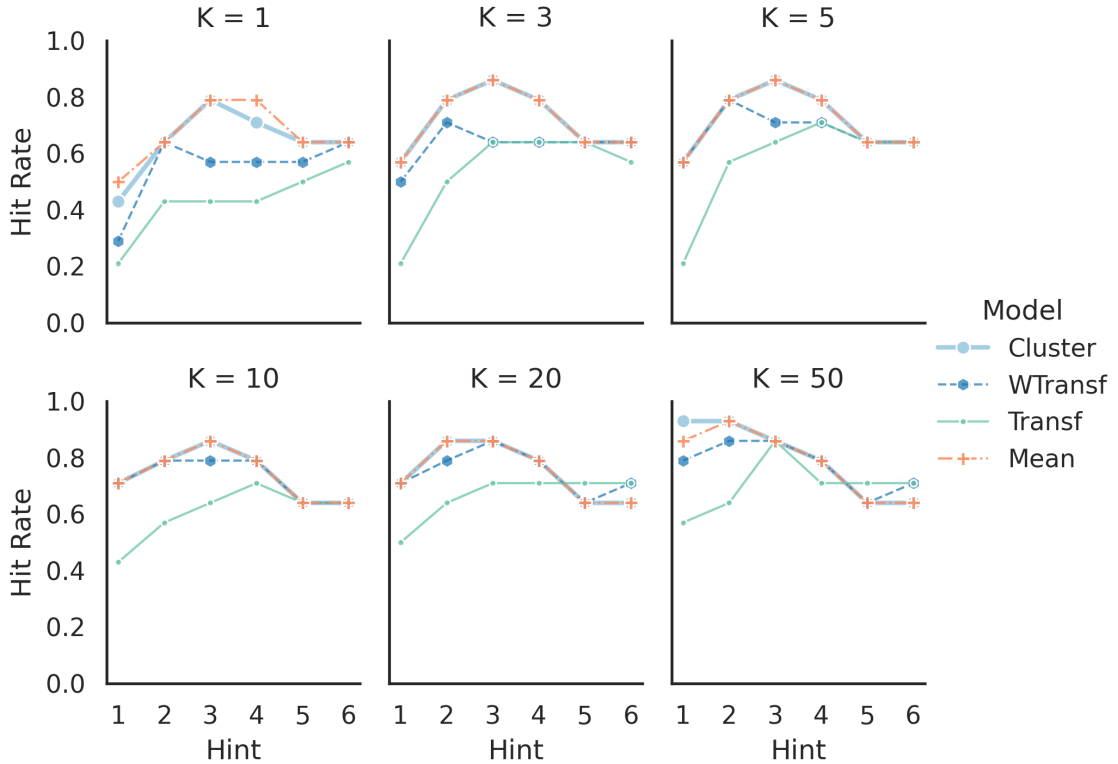


Figure 6.2: Hit Rate at K at different hints on four approaches.

follows:

- **Mean:** the mean pooled embedding was used as the global event embedding. No training was required.
- **Cluster:** OPTICS clustering was applied for each scene with `max_eps=0.5` and `min_samples=2`
- **Transf:** *one* layer of PyTorch¹ implementation of Transformer Encoders was trained for 10 epochs using the captions described in LLQA dataset [204] with `n_head=8` and `d_model=1024`. The outputs were mean pooled.
- **WTransf:** same settings with Transf. The outputs were used to create a weighted mean embedding.

Surprisingly, the straightforward method of mean pooling achieves the highest results in most cases. As for clustering, not only the search space has increased, but the hit rates at $K = 1$ are also slightly lower. Furthermore, despite having more parameters, both the weighted mean (WTransf) and averaging the output (Transf) from Transformer encoders produce generally worse performance, especially at lower values of K and when

¹<https://pytorch.org/>

Table 6.1: Mean $H@K$ for LSC’22 queries using Mean Pooling. I am most interested in the modified version of H@3 because (i) once the user find the correct answer, more hints are not needed and (ii) the user interface can display three events at a time.

Hint	H@1	H@3	H@5	H@10	H@20	H@50	Mod H@3
1	0.50	0.57	0.57	0.71	0.71	0.86	0.57
2	0.64	0.79	0.79	0.79	0.86	0.93	0.79
3	0.79	0.86	0.86	0.86	0.86	0.86	0.86
4	0.79	0.79	0.79	0.79	0.79	0.79	0.86
5	0.64	0.64	0.64	0.64	0.64	0.64	0.86
6	0.64	0.64	0.64	0.64	0.64	0.64	0.86

fewer hints are used. This could be explained by the limited size of the training dataset, which contains only 13,317 captions.

The best-performing setting is recorded in Table 6.1. From the experiments, it was observed that more hints do not mean better results. In fact, the system performed the best when 2–3 hints were given, without ‘before’, ‘after’, or misleading hints (for example wrong year). Thus, a modified H@K metric, denoted as Mod H@K, is also reported in the table, where $H@K(i) = \max(H@K(i), H@K(i-1))$ to account for the fact that more searches are not needed after the correct submission has been made in the live LSC event. It is also important to be aware of the trade-offs when choosing to show events instead of individual images when it comes to the amount of results that can be effectively be displayed on the user interface. MyEachtra’s event-based user interface can fit 3 events at most, thus I am most interested in Mod H@3, when the assumption of *no scrolling is needed* is made. Here, the table suggests that the answers for 57% (8 out of 14) of the queries can be found using only the first hints. With more hints, the user can find the correct image in 86% of the queries (12 out of 14).

6.3 User Study: Comparison with Myscéal

In order to compare the performance of MyEachtra with the previous image-based Myscéal system, a user study was conducted with eight participants. The criteria for selecting the participants were mentioned in Chapter 3. Each participant was asked to complete eight KIS queries from LSC’22, with four queries from each system. Evaluating the performance of MyEachtra for ad-hoc queries was not possible because the groundtruth image sets were not released by the organisers. The participants were asked to complete the queries in a fixed order, with the system alternating between MyEachtra and Myscéal decided

by the Latin Square design in Table 6.2. Participant IDs were randomly assigned to each participant to reduce bias. Following this design, each query is completed by four participants using MyEachtra and four participants using Myscéal. To eliminate the effect of using different embedding models, the same CLIP model was used for both systems.

Table 6.2: Latin Square design for the user study to evaluate the event-based MyEachtra system on LSC’22 KIS queries. A and B represent MyEachtra and Myscéal respectively. Q1–Q8 represents the eight queries.

Participant ID	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
1	A	A	A	A	B	B	B	B
2	A	A	A	B	B	B	B	A
3	A	A	B	B	B	B	A	A
4	A	B	B	B	B	A	A	A
5	B	B	B	B	A	A	A	A
6	B	B	B	A	A	A	A	B
7	B	B	A	A	A	A	B	B
8	B	A	A	A	A	B	B	B

Table 6.3: Statistics of each system’s performance in the user study for KIS queries in LSC’22.

System	Score (Mean)	Wrong submissions (Mean)	Solved queries (Sum)	Time taken (Mean)
MyEachtra	76.58	1.34	30	61.89
Myscéal	68.64	1.03	26	49.31

The results are shown in Figure 6.3 and summarised in Table 6.3. For this user study, the scoring formula from the LSC campaigns, described in Chapter 3, was reused. It considers the time of submission and penalises wrong submissions. The score for each query is shown in the left box-plots. No metrics show a significant difference between the two systems in this user study. However, this still suggests that MyEachtra is a competitive system, as it has a higher mean score and a higher number of solved queries. A close look at the performance of each user can be seen in the right chart, where the mean score of MyEachtra is higher than Myscéal for four users (User 1, 6, 7, 8) and the reverse is true

for the other four users (User 2, 3, 4, 5). This is due to the varied difficulty of the queries and the design of the Latin Square. In other words, users with the opposite settings (for example User 1 vs 5, User 2 vs 6) tend to have opposite performance trends. It is worth noting that despite these differences, the overall performance of MyEachtra is still higher than Myscéal.

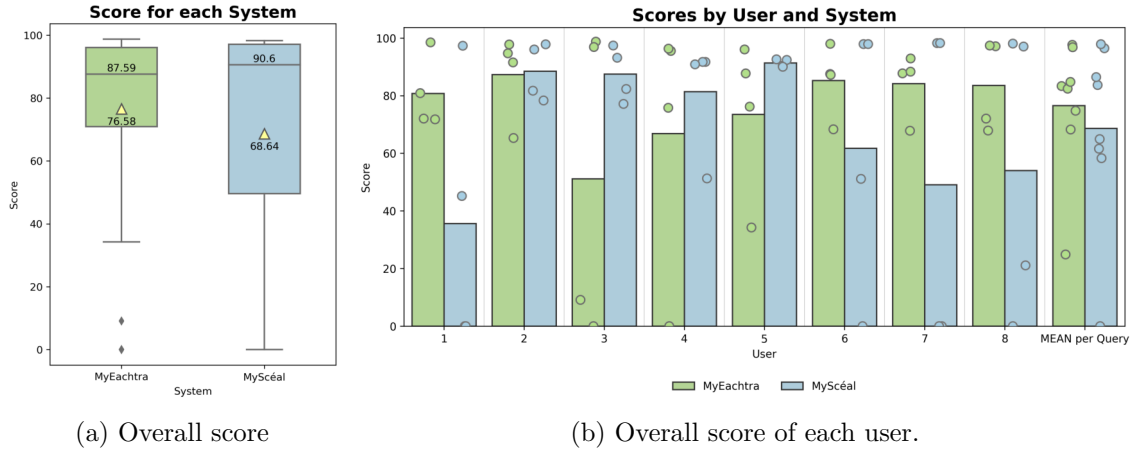


Figure 6.3: Comparison between MyEachtra and E-Myscéal for LSC'22 queries.

6.4 MyEachtra at LSC'23

Although the main focus of this dissertation is QA for lifelog retrieval, I believe a well-rounded system should be able to solve all KIS, Ad-hoc, and QA tasks. In this section, KIS and Ad-hoc queries in LSC'23 are used to evaluate the performance of MyEachtra.

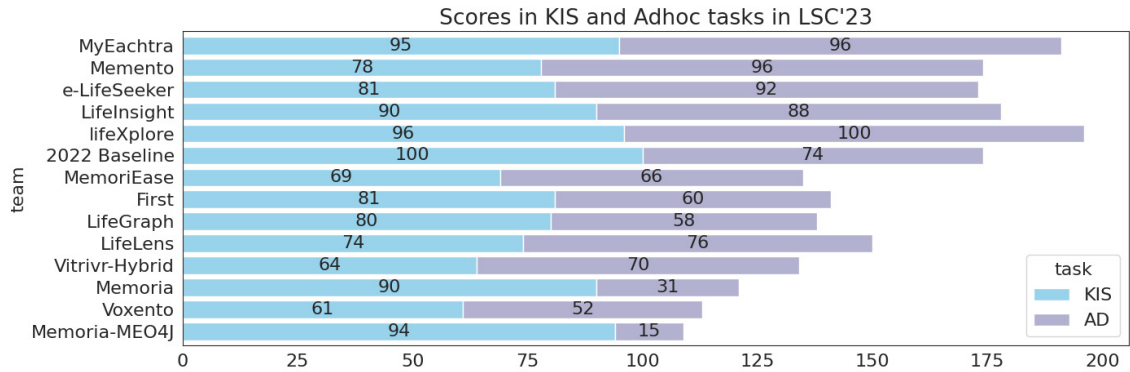


Figure 6.4: Overall score of all teams in LSC'23. The baseline system is E-Myscéal.

Six queries were proposed by the organisers of the LSC'22 campaign for each of the tasks in the expert run. The overall scores of all teams can be seen in Figure 6.4. MyEachtra ranked third and second respectively for KIS and Ad-hoc tasks. Compared with the baseline system E-Myscéal (as depicted as 2022 Baseline in the charts), MyEachtra slightly

fell short in KIS tasks (p-value nearly 0) but significantly outperformed in Ad-hoc tasks (p-value=0.01). Summing both KIS and Ad-hoc scores, MyEachtra achieved the second-highest overall score (191) after lifeExplore [187] (196).

It is worth noting that six out of 14 teams managed to solve all the KIS tasks, five of the rest solved five tasks, and the remaining three solved four tasks. It is a significant improvement for all teams compared to LSC’22. This is shown in Figure 6.5. Among the teams, MyEachtra solved six tasks but submitted one irrelevant image. E-Myscéal also solved six tasks with two wrong submissions. However, even though MyEachtra was the third-fastest team to solve the tasks (Figure 6.6), its speed was still slightly slower than E-Myscéal. Thus, E-Myscéal achieved the highest score in KIS tasks. This is consistent with the results of the user study, where E-Myscéal was faster than MyEachtra.

With respect to Ad-hoc tasks, MyEachtra’s recall rate was only 0.31, which put it in fifth place. However, MyEachtra shared the highest precision of 0.84 with lifeExplore [187]. This is shown in Figure 6.7. As a result, MyEachtra achieved the second-highest score in Ad-hoc tasks, only behind lifeExplore, shared the same score with Memento [5], and significantly outperformed E-Myscéal (p-value=0.01) due to the higher precision.

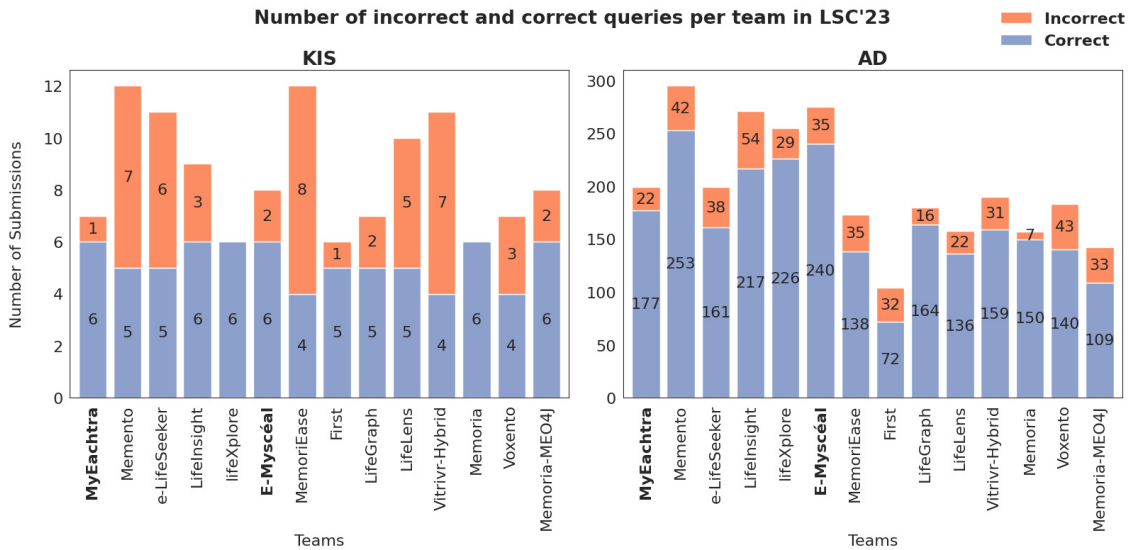


Figure 6.5: Number of incorrect and correct submissions of teams in LSC’23.

6.5 Discussion

In this section, I discuss the results of the user study and the LSC’23 evaluation. I also highlight the limitations of MyEachtra and propose some future work. Overall, the performance of MyEachtra is complementary to that of E-Myscéal, which means it performs well where E-Myscéal falls short. For example, E-Myscéal is exceptional at solving the task quickly, but the tradeoff is that it is more likely to submit wrong images. On the

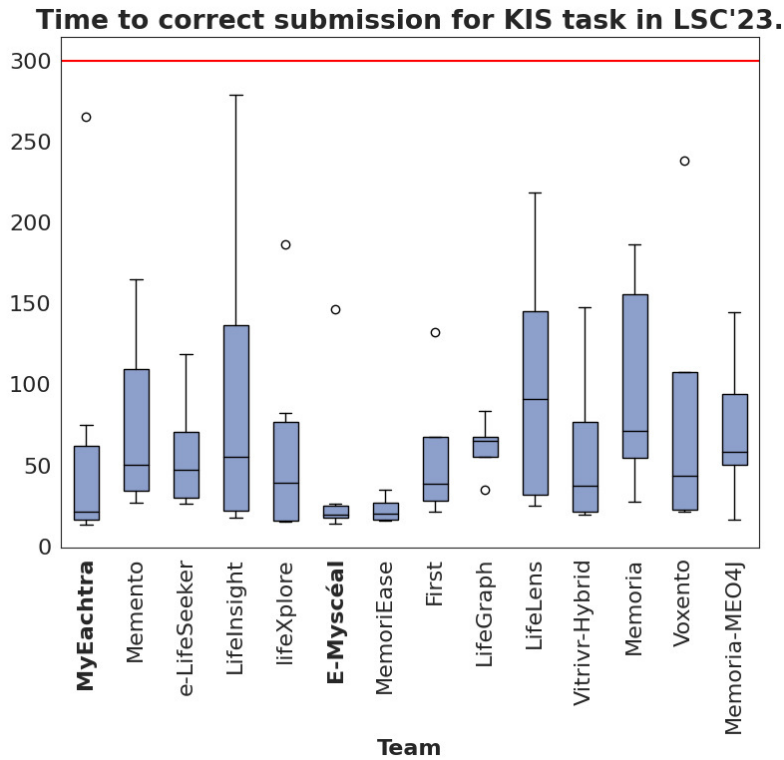


Figure 6.6: Time to first correct submission all teams in LSC'23 in KIS tasks. Ad-hoc tasks are not included because the time does not matter in this task.

other hand, MyEachtra is more accurate but slower. However, even though MyEachtra is slower, it is still very fast compared to other teams in LSC'23. As a result, MyEachtra is able to achieve the second-highest overall score in LSC'23.

Although E-Myscéal has higher precision in the user study, it is worth noting that the user study was conducted with a small number of participants, with a small number of submissions (due to the nature of KIS tasks). Looking at the results of LSC'23, MyEachtra has both higher precision and recall than E-Myscéal. This can be explained by some of the design choices of MyEachtra. Firstly, the event-based interface groups images based on their temporal proximity, which can provide more context so that the user can make more informed decisions. Moreover, for each event, if the user submits one image, the other images in the event are also likely submitted and correct. Additionally, the redesigned user interface avoids the repetition of the same event in different areas of the result list, which can increase the complexity of the task and lead to more wrong submissions if the user is distracted. However, the downside of MyEachtra is that the user has to scroll more as the number of images shown is reduced (as seen in Figure 6.1). This is reflected in the user study, where MyEachtra took longer to complete each query.

On the note of submission speed, E-Myscéal has the advantage of having a traditional left-to-right, top-to-bottom layout, which is familiar to most users. On the other hand,

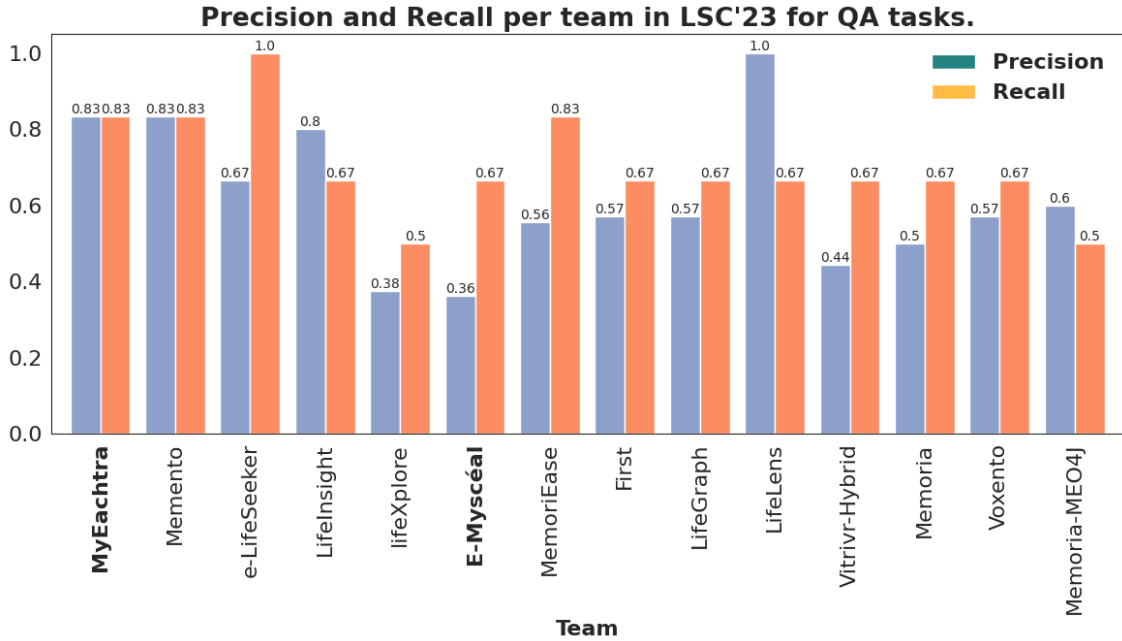


Figure 6.7: Precision and Recall for Ad-hoc tasks of all teams in LSC'23.

even though MyEachtra has a more intuitive interface, it is still a new design that users need to get used to. User experience and preference are also important factors that can affect the performance of the system. In the user study, I found that some participants preferred the interface of MyEachtra while others preferred E-Myscéal. All in all, despite not having a clear winner, the results of the user study and LSC'23 evaluation show that MyEachtra is a promising system that can complement E-Myscéal and achieve a higher overall score. In the grand scheme of addressing the lifelog question answering task, MyEachtra's event-based approach is a step towards a unified pipeline for lifelog retrieval systems.

As the first system to use an event-based approach, MyEachtra has some limitations. Firstly, the event segmentation process is still very simple and most likely not optimised. Adaptive segmentation based on the query is a possible solution to this problem, as explored in [62]. In this approach, the segmentation is based on the relevance of the images to the query, and the Euclidean distance between consecutive images based on the relevance scores. The images are then grouped into events based on the distance between them. This approach was adapted to Myscéal and MyEachtra by replacing the relevance score with the visual embedding and the Euclidean distance with cosine distance, thus removing the dynamic nature of the segmentation. However, more sophisticated segmentation methods can be explored. For example, the current location-based segmentation can be replaced a time-based segmentation if the query contains a time hint. Secondly, the event-based approach is not suitable for all queries. For example, if the query is about a specific ob-

ject, the user may not be interested in the events themselves. Thirdly, aggregating image features into event features is still a challenging task. The current approach of averaging the image features is simple and effective, but it is not optimal as fine-grained information can be lost and the order of images is not considered. In the future, I plan to explore more advanced techniques to aggregate image features into event features. Finally, the current user interface is still not without flaws, as it limits the number of images that can be shown in an event and requires much more scrolling to see more results. Future works can explore more effective ways to display events and images.

6.6 Conclusion

On this chapter, I presented modifications to Myscéal that shift the focus of lifelog retrieval from images to events, aiming to move towards a unified model for lifelog question answering. Our new system, MyEachtra exploits pretrained image-text models to create event embeddings. Experiments were conducted, and the results suggested averaging image embeddings to create event embeddings is the most suitable approach for MyEachtra at this stage, resulting in a reduced search space without sacrificing performance. Additionally, the user interface was readjusted to show relevant events and focus on contextual information effectively. To evaluate the performance of MyEachtra, a user study was carried out to compare it with the previous image-based E-Myscéal system. Furthermore, MyEachtra was also evaluated in LSC'23, where it achieved the second-highest overall score and had competitive performance compared to E-Myscéal in KIS tasks and significantly outperformed E-Myscéal in Ad-hoc tasks. In conclusion, the Research Question 3.1, *Does the event-based retrieval support the user to achieve comparative performance to image-based retrieval for lifelog data?*, is answered positively. The next chapter of this dissertation will address the final research question by incorporating question answering into MyEachtra to create a unified lifelog retrieval system.

Chapter 7

Lifelog Question Answering System

This chapter is dedicated to addressing Research Question 3.1, which is *Can the tailored question answering approach improve the performance of interactive lifelog retrieval?*. This is the final step of the proposed pipeline for integrating question answering capabilities into lifelog retrieval systems. The goal of this step is to evaluate the performance of the proposed pipeline in interactive lifelog retrieval tasks. To achieve this goal, I incorporated question answering models into the MyEachtra system described in Chapter 6. The resulting system was evaluated in a user study to compare its performance with the baseline system, E-Myscéal [207], which was the best-performing system in the LSCs for three years in a row. In this study, inspired by the lifelog question answering tasks in the LSCs, I also designed a new set of questions, detailed in Section 7.2.1, in order to evaluate the performance of the systems in answering general lifelog questions, as opposed to the specific questions with context in the LLQA dataset. Moreover, MyEachtra’s performance in LSC’23, the first lifelog challenge that includes a text-based QA task, is reported in Section 7.4. The chapter is then concluded with a discussion of the results and future works in Section 7.5.

7.1 Lifelog Question Answering Pipeline

Inspired by the open-domain QA pipeline [29], I formally propose a pipeline for the lifelog QA system as shown in Figure 7.1. Two key components of the pipeline are (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.

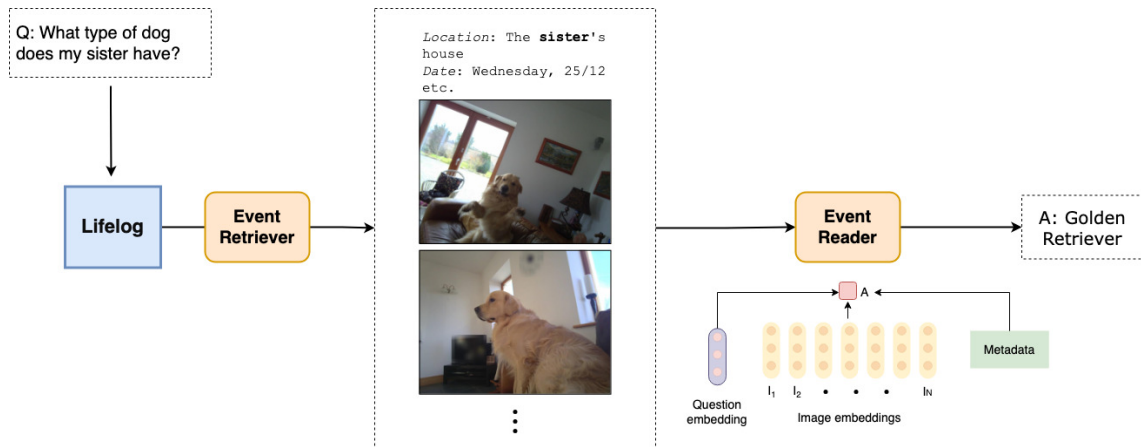


Figure 7.1: The proposed pipeline for the QA system. The Event Retriever is in charge of retrieving the lifelog data that are relevant to the given question. On the other hand, the Event Reader component is responsible for generating answers based on the retrieved data.

7.1.1 Event Retriever

The first component is a crucial part of the pipeline, as it is responsible for retrieving the relevant lifelog information that are 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 the literature review in Chapter 2.

To adapt conventional image-focused lifelog retrieval systems, a post-processing step might be useful to aggregate the information from the retrieved data and reduce the volume of the necessary 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 [213], 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 [167] 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, I 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.

7.1.2 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, I 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, I 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.

FrozenBiLM [229] is employed as the VideoQA model, which builds on a frozen bidirectional language model as well as a frozen visual encoder, CLIP [167]. FrozenBiLM was pre-trained on a large-scale video-caption pairs dataset WebVid10M [14]. 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 [204], 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 [236] instead.

The new addition to the model is the use of the text-only QA model to handle non-visual information. Although FrozenBiLM is capable of handling metadata in the form of

text, its approach of predicting the most probable answer from a large set of candidates is ineffective as most answers concerning lifelog metadata are very specific and personal. For example, the answer to the question ‘Where did I have dinner last Thursday?’ is likely to be a specific restaurant name that is not included in a pre-defined set of candidate answers. Therefore, a text-only QA model is used to handle these types of questions. Specifically, RoBERTA [127], a pretrained language model, is used to generate the answers. Related information from the metadata is used to generate a contextual prompt 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>’. RoBERTA is used to predict the answer span from the generated prompt, thus is able to handle questions that require specific non-visual information, such as location and time.

7.2 User Study Setup

To evaluate the effectiveness of the proposed lifelog QA system, a user study was conducted, 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.

7.2.1 Lifelog Questions

In order to compile a comprehensive QA dataset, I utilised the largest two lifelog datasets in the LSC, namely LSC’21 [69] and LSC’22 [70]. 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 days’, 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 (for example restaurant) where some specific events happened. For example, ‘Where did I go to 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 Figure 7.2. 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_u1v3I0dgU4wU.

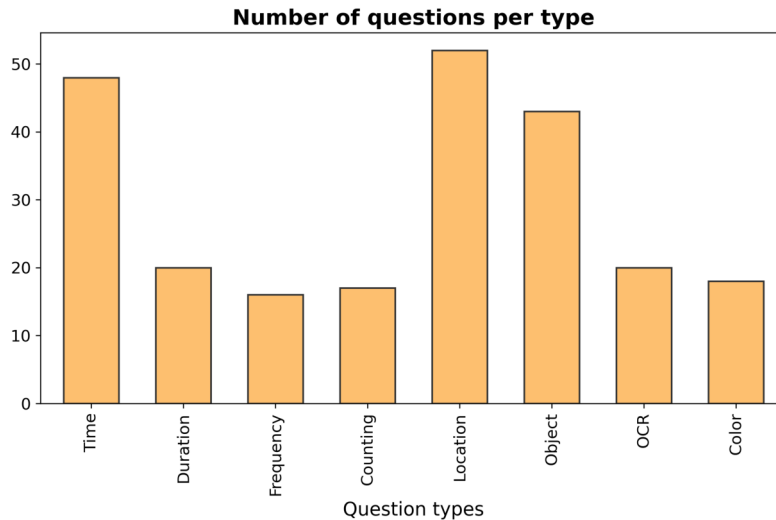


Figure 7.2: Distribution of the questions in the test collection.

7.2.2 User Study Design

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 allocated 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. The QA group was asked to use the QA system first, then the baseline system. This is to ensure that the order of the systems does not affect the results.

Each participant had a training period of 10–15 minutes 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 *eight* 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 is correct or not. If the answer is incorrect, the participants were asked to try again. If they could not find the answer within 3 minutes, 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) [109], 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). The participants were also asked to provide feedback on the system, which will be used to improve the system in the future. The questionnaire and the QA tasks are shown in Appendix A.

Taken from the LSC, the performance of the systems is measured based on (1) the

accuracy of the answers, (2) the number of wrong submissions, and (3) the time taken to answer the questions. For each task, if it is solved (the correct answer was submitted), the score is calculated as follows:

$$\text{score} = 100 - 50 \times \frac{\text{time taken}}{180} - 10 \times \text{number of wrong submissions} \quad (7.1)$$

If the task is not solved, the score is 0.

7.2.3 Baseline System

The baseline system used in this user study is the state-of-the-art lifelog retrieval system that was described in Chapter 4, which was developed for the LSC prior to 2023, E-Myscéal [207]. This is also the baseline system for the LSC’23 [71].

As a reminder from Chapter 4, E-Myscéal is a lifelog retrieval system that 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 [175], while the mechanism to retrieve the visual data field is the cosine similarity between the query embedding and the image features. The query and image features are extracted using the OpenAI’s CLIP model [167].

The user can also browse the lifelog images using a popover timeline, which is shown when the user clicks on any image shown in 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.

7.2.4 QA system

We use the proposed pipeline to integrate QA capabilities into the E-Myscéal system by (1) shifting the unit of retrieval to events, which is the main difference between MyEachtra and E-Myscéal [209] in the retrieval stage; and (2) adding a QA component to generate the answers based on the retrieved events. Refer to Section 7.1 for more details about the pipeline.

In MyEachtra, the question is used directly as a search query to find the most relevant events. The top-10 events are passed through FrozenBiLM and RoBERTA to get potential answers, which are shown on the left panel of the user interface in Figure 7.3. Nonetheless,

users can choose to run the model again for any event they find interesting by clicking on the QA button underneath each row. Once an answer is found, users have the option to swiftly copy it by clicking on it and then pasting it into the submission input field located in the bottom left corner. Alternatively, if they can deduce the answer from the displayed events, they can manually type it into the box and submit it.

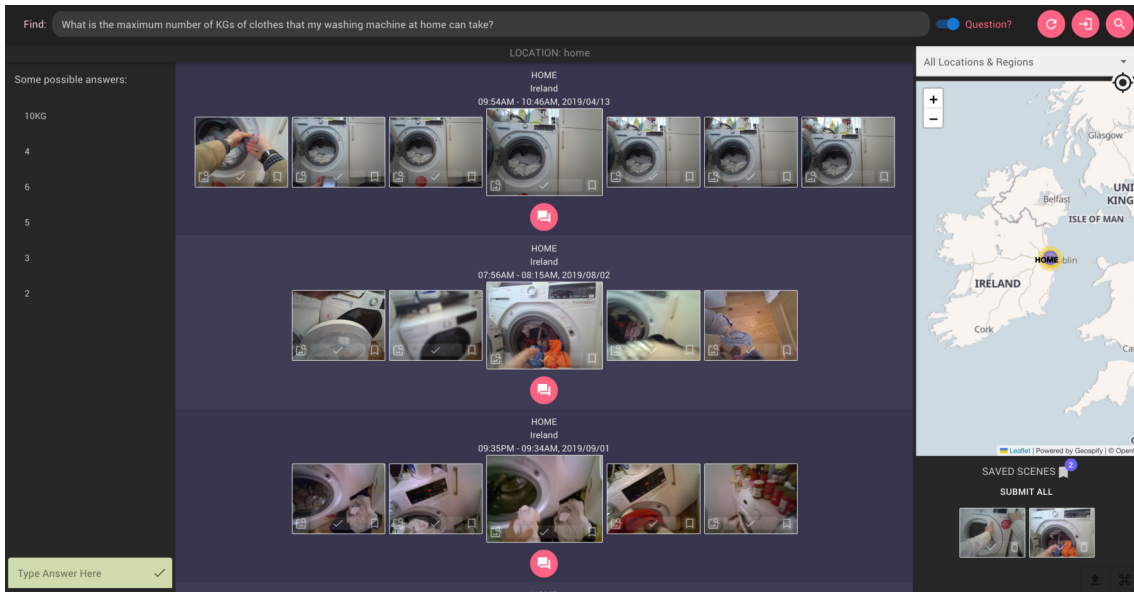


Figure 7.3: MyEachtra’s user interface. For non-QA tasks, the left panel is hidden.

7.3 User Study Results

7.3.1 Overall Score

The overall score of each system is calculated as the average score of all the tasks. The results are shown in Figure 7.4. The QA system has a higher average overall score than the baseline system. 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 seconds, while that of the baseline system is 74.78 seconds. The performance of each user is also shown in Figure 7.5. It is not clear that the QA system is better than the baseline system, as the performance of the two systems is not significantly different.

To understand the performance of the systems under each type of question, the scores of the questions of each type are calculated. The results are shown in Figure 7.6. The QA system has higher average scores than the baseline system in terms of the overall score for Location, Object, and Counting questions, and lower average scores for Frequency

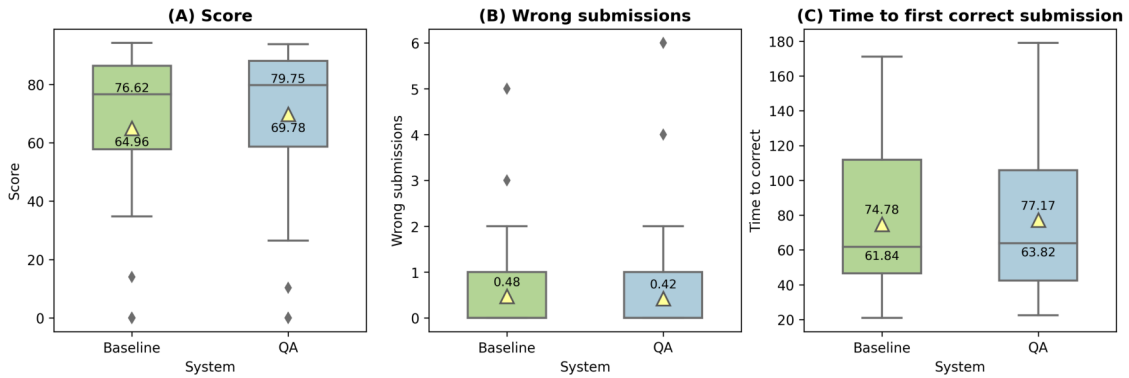


Figure 7.4: (A) Overall score and (B) Time taken to answer the questions of the two systems.

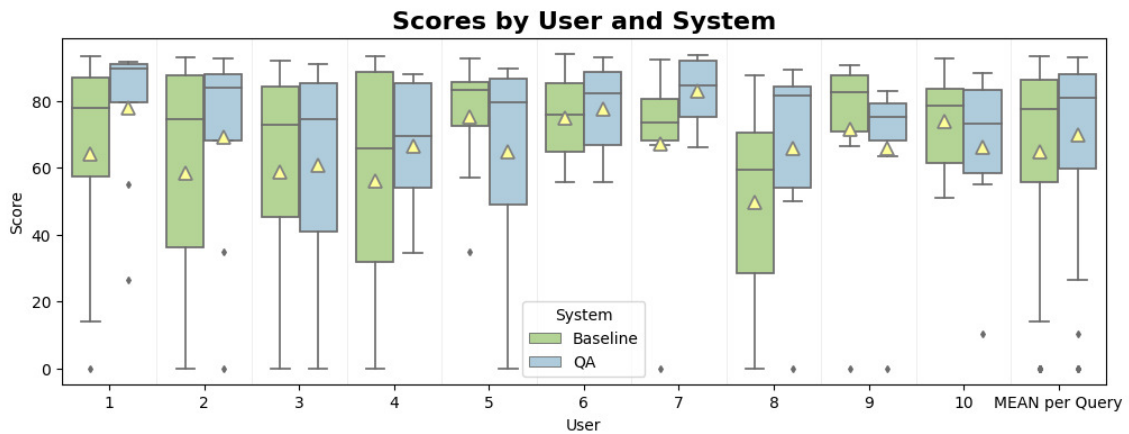


Figure 7.5: Overall score of each user.

questions. The average scores of the two systems are not significantly different for Time, OCR, Colour, and Duration questions.

7.3.2 Importance of Experience

The results show that the QA system has better scores than the baseline system in terms of the overall score. However, the performance of the QA system is not much better than the baseline system. This is possibly because the participants have very little experience with lifelogging and question answering. To have a better understanding of how the users perform with more experience, the average scores of the first system and the second system used by each user are examined. The results are shown in Table 7.1. 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 suggests that the QA

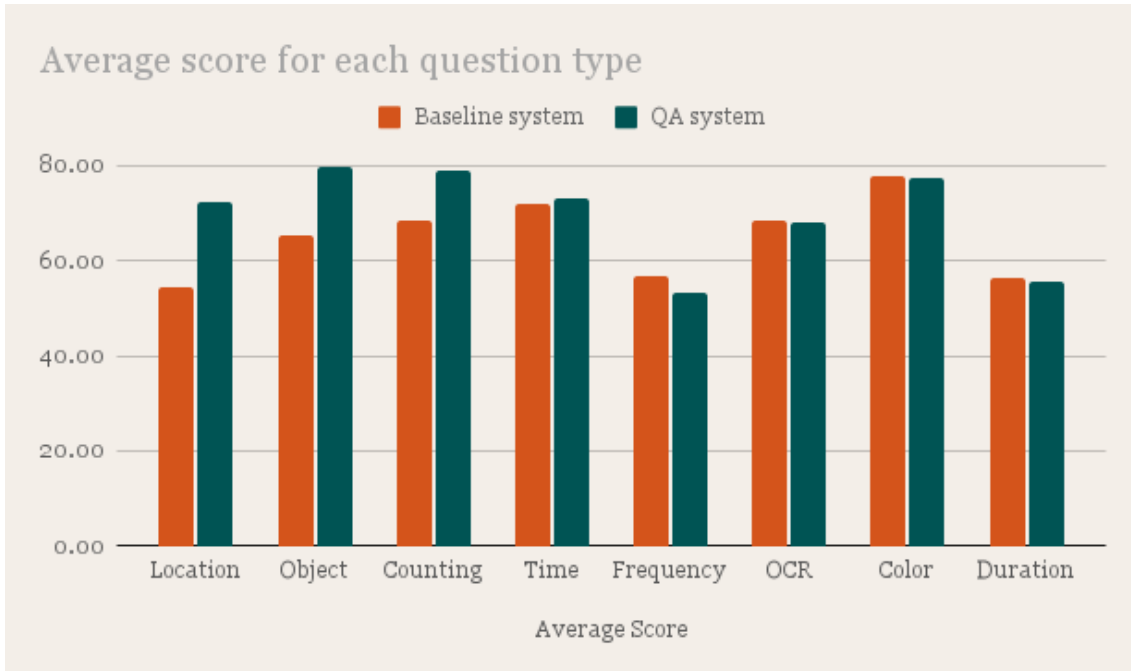


Figure 7.6: Average scores across different types of questions.

system could be easier to use than the baseline system. Considering the second system only, the difference between the two systems is not clear (68.12 for the baseline system and 68.00 for the QA system). This possibly is because the users are already familiar with the tasks.

Table 7.1: Average score of the first system and the second system used 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

7.3.3 User Experience Questionnaire

Figure 7.7 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 Figure 7.7, 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.

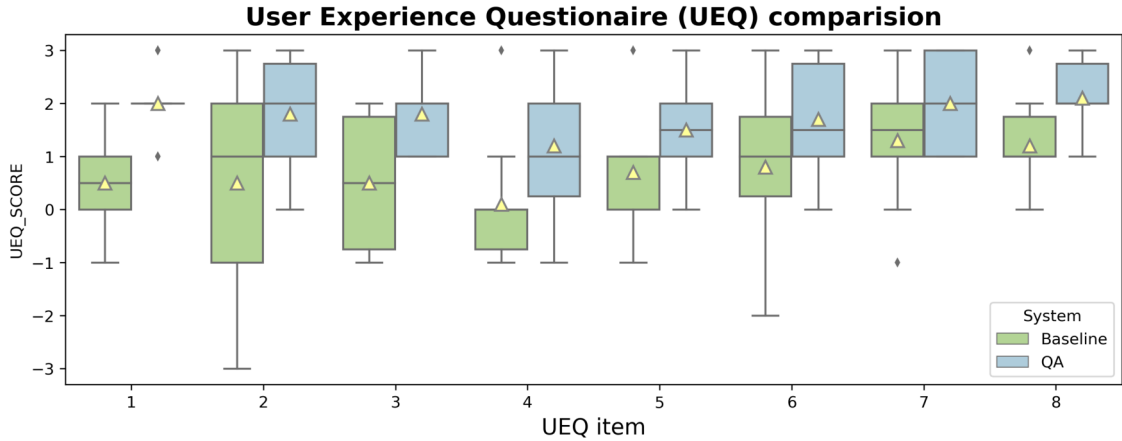


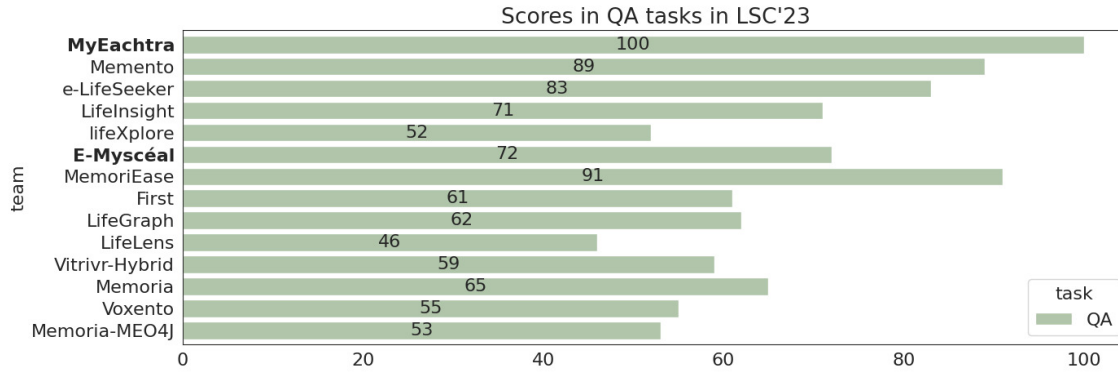
Figure 7.7: Results of the User Experience Questionnaire. MyEachtra significantly outperforms the baseline system in all aspects (p-value is nearly 0).

7.4 MyEachtra In LSC’23

In this section, we further analyse MyEachtra’s performance in six QA tasks in the expert run of LSC’23 [71]. The results, as shown in Figure 7.8, underscore MyEachtra’s achievements, securing the highest score in QA tasks, followed by MemoriEase [216] and Memento [5]. Moreover, the baseline system, E-Myscéal [207], achieved the fifth spot in the ranking, which is expected given its competitive performance in the LSC but its lack of support for QA tasks. Therefore, E-Myscéal only achieved 71% of the score of MyEachtra, highlighting the significance of QA capabilities.

Taking a closer look at the precision and recall metrics in Figure 7.9, MyEachtra submitted five correct answers and one incorrect submission, resulting in one unsolved QA task. This gave MyEachtra a precision and recall of 0.83 each, marking the second-highest precision and recall among all participating teams. However, the overall score of MyEachtra was still the highest, as it was the fastest team to submit the correct answers. Comparatively, MemoriEase, the second-ranking team, also managed to submit five correct answers but with four incorrect submissions, resulting in a lower precision score of 0.56. Meanwhile, the third team in the ranking, Memento, mirrored MyEachtra’s performance in this metric category but was compromised in terms of speed, as discussed in the next paragraph and illustrated in Figure 7.10. Returning to the precision and recall metrics,

Figure 7.8: Overall score of all teams in LSC'23 for QA tasks. MyEachtra achieved the highest score. The baseline system, E-Myscéal, achieved the fifth spot in the ranking with 71% of the score of MyEachtra (p-value = 0.0007 for the average score per question).



E-lifeseeker [152] was the only team that solved all six tasks (highest recall). However, its three incorrect answers resulted in a lower overall score (fourth place) than MyEachtra. Similarly, Lifelens [83] achieved a perfect precision score, yet only managed to solve four tasks, affecting its overall performance. Finally, despite solving four tasks, E-Myscéal exhibited the lowest precision score of 0.36 with seven incorrect answers, positioning it mid-tier in the overall ranking.

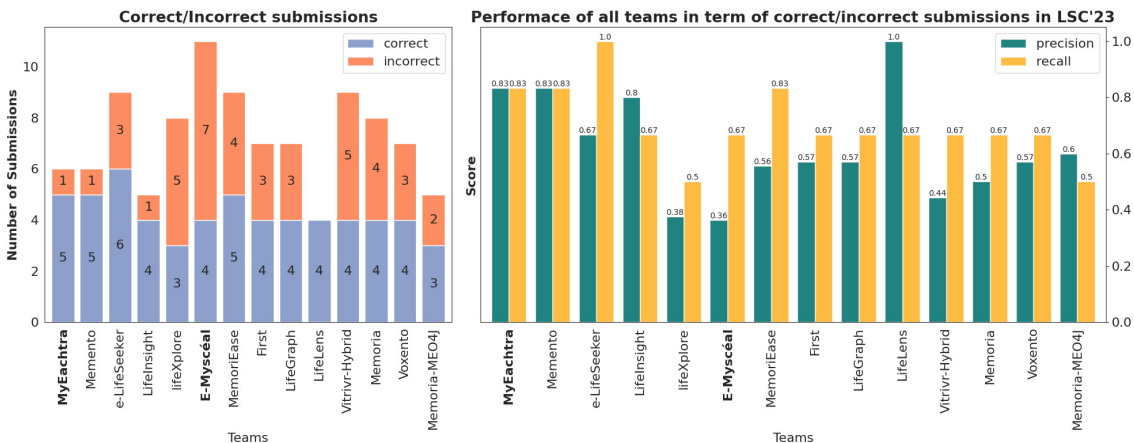


Figure 7.9: Accuracy of submissions of all teams in LSC'23.

Evaluating the time taken to submit the first correct answer, MyEachtra significantly outperformed all other teams (p-value = 0.04), as illustrated in Figure 7.10. The average time taken to solve the task for MyEachtra was 59.39 seconds, with a median of 50.67 seconds. This put MyEachtra 1.5 times faster than the runner-up team, Memento, whose average time stood at 91.14 seconds. Notably, that of the E-Myscéal was commendable at 89.85 seconds, which is still considered fast compared to other teams. This indicates that MyEachtra is a fast and accurate system for lifelog QA tasks.

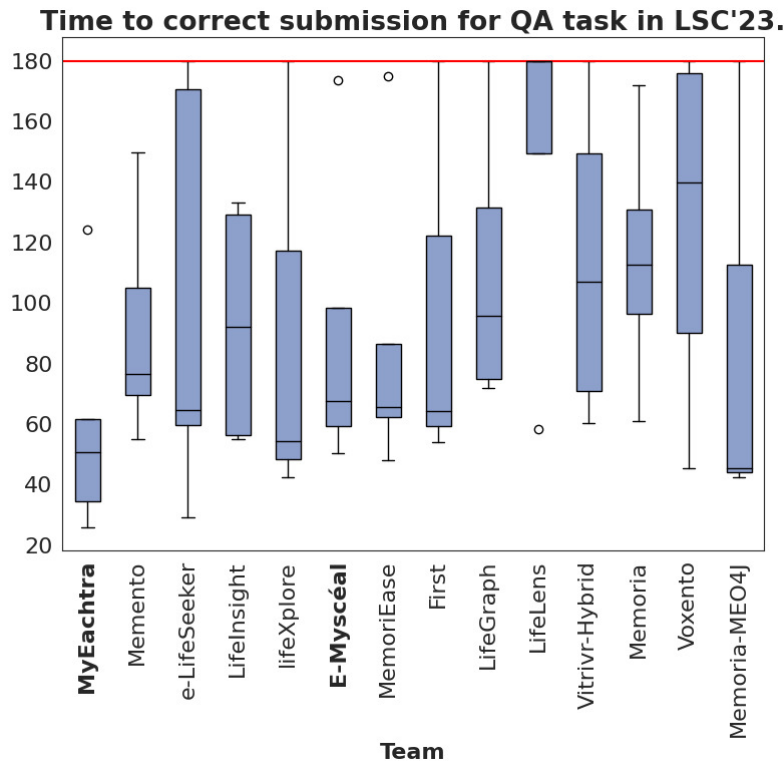


Figure 7.10: Time to first correct submission all teams in LSC'23 in QA tasks.

7.5 Discussion

In this section, I discuss the outcomes of the user study and the LSC'23 evaluation for the QA system, MyEachtra, in various aspects and highlight the insights gained from the results. Additionally, based on the lessons learned, I propose some future work to improve the system.

First of all, the user study revealed that the QA system had a higher overall score than the baseline system. However, MyEachtra's performance did not show a significant advantage over the baseline system. It is worth noting that the study participants had limited experience with lifelogging and question answering. With more experience, MyEachtra's performance could improve, as indicated by the expert run of LSC'23 just discussed in Section 7.4. MyEachtra stood out as the fastest system in LSC'23, a crucial aspect of this benchmarking challenge. In addition, MyEachtra achieved the second-highest precision and recall in LSC'23. This suggests that MyEachtra is both fast and accurate for lifelog QA tasks.

The main difference between MyEachtra and the baseline system lies in MyEachtra's event-based approach. This approach was inspired by the observation that information needs are more likely to be addressed by an event rather than a single image, as many lifelog search queries have multiple images as the ground truth answer. Consequently, I believe

that an event-centric approach is better suited for QA tasks, a belief supported by the results of the user study and LSC'23. The event-based approach also enables MyEachtra to handle more complex queries, such as those related to duration and frequency, which are challenging to answer without any organisation of the lifelog data. However, the system's performance could be further enhanced by implementing a dynamic event segmentation instead of the fixed event segmentation employed in MyEachtra. This is a potential area for future work for MyEachtra.

During the development of the questions for the user study, I encountered the challenge of designing unambiguous questions that could be easily evaluated. This difficulty arises from the immense diversity and complexity of lifelog data. Moreover, some questions have multiple plausible answers, making evaluation a complex task. For instance, a question like 'What did I eat for breakfast on the 1st of January 2020?' can have acceptable answers like 'eggs' or 'bacon' if the lifelogger had a full Irish breakfast. A human judge is required to evaluate the answers, which is not scalable. Therefore, there is a need for a more comprehensive evaluation framework that can automatically assess responses, accounting for multiple potential answers, in future research.

The research question addressed in this chapter is *Can the tailored question answering approach improve the performance of interactive lifelog retrieval?*. Compared to the baseline system, E-Myscéal MyEachtra achieved a higher overall score in the user study and the LSC'23 evaluation in the QA tasks, thus demonstrating the effectiveness of the proposed pipeline for integrating question answering capabilities into lifelog retrieval systems. Therefore, I conclude that the proposed pipeline is a viable solution to improve the performance of lifelog retrieval systems.

7.6 Conclusion

This chapter presented a novel pipeline for integrating question answering capabilities into lifelog retrieval systems. Our approach enables users to ask questions about their lifelogs and receive text-based answers, thereby enhancing the effectiveness and user satisfaction in lifelog retrieval tasks. The results of the user study demonstrated the superiority of our proposed system over the baseline approach, with significant improvements in both effectiveness and user satisfaction metrics. Furthermore, the results suggested that our proposed system is particularly well-suited for new users, offering a more intuitive and efficient lifelog retrieval experience. Compared to the other systems, our proposed system achieved the highest score in the LSC'23 expert run for question answering tasks, with the second-highest precision and recall. This indicates that our proposed system is a fast and accurate solution for lifelog question answering tasks. In the future, I plan to explore the use of dynamic event segmentation to improve the performance of the system and adapt to the different types of questions. I also plan to develop a more complete

evaluation framework to evaluate the answers automatically, which needs to take into account multiple possible answers.

Chapter 8

Conclusion

This chapter addresses the answers to the research questions and how the hypothesis is supported by the proposed approaches. The contributions of this thesis are also summarised in this chapter. Lastly, I will discuss the limitations of the proposed approaches and suggest some possible directions for future works.

8.1 Hypothesis and Research Questions

The focus of this thesis is to explore approaches to question answering from lifelog data as well as to incorporate these approaches into interactive lifelog retrieval systems. In this section, I will revisit the hypothesis and research questions, and examine them with respect to the proposed solutions and the experimental results.

Hypothesis *Question Answering techniques can improve upon state-of-the-art interactive retrieval systems for lifelog data by improving the result's quality and supporting quick access to relevant information.*

Several related research questions were developed to guide the research process, as follows:

Research Question 1 (RQ1). **How to design a state-of-the-art interactive lifelog retrieval system that assists a novice user to quickly locate items of interest from a conventional multimodal lifelog?**

This research question focuses on the development of an interactive lifelog retrieval system with novice users in mind by incorporating a simple, straightforward interface and intuitive interaction mechanisms while maintaining the system's performance. To answer this question, I proposed a novel interactive lifelog retrieval system, Myscéal and evaluated its performance against other lifelog systems in benchmarking programmes as well as user studies. The results showed that Myscéal outperformed other systems in terms of speed and accuracy while maintaining a simple and intuitive interface. Notably, in the three

annual iterations of the Lifelog Search Challenge (LSC)[68–70], Myscéal was the winning system in overall performance. Findings from the user studies further suggest that the system can support novice users to effectively interact with their lifelog data. This implies that the incorporation of user-friendly features and intuitive interaction mechanisms enhances result quality and user engagement, aligning with the hypothesis. Thus, E-Myscéal, which is the latest iteration of the Myscéal system, was used as the baseline system for the subsequent research questions.

Research Question 2 (RQ2). How can we evaluate different approaches to question answering on lifelog datasets?

The research question requires a two-step approach to answer. Therefore, I proposed two sub-questions:

RQ2.1. How to adapt existing lifelog test collections to evaluate approaches to lifelog question answering? This research question focuses on the development of a lifelog QA dataset, which is necessary to evaluate the effectiveness of various QA approaches. To answer this question, I extended the existing Lifelog Search Challenge (LSC)[68] lifelog collection by adding over 15,000 multiple-choice and yes-no questions. The resulting QA dataset, LLQA[204], is the first lifelog QA dataset that is publicly available and also comes with a set of valuable lifelog captions.

RQ2.2. What existing question answering techniques are most effective when applied to lifelog data? A pilot study was conducted to establish a human gold standard accuracy for the dataset, along with state-of-the-art text QA and video QA models at that time. A more comprehensive benchmarking study was conducted to evaluate the performance of multiple QA techniques in the relevant domains, especially video QA due to the similarities between lifelog and video data. With a focus on pretrained vision-language models, the benchmarking study evaluated the performance of pretrained multimodal models fine-tuned on the LLQA dataset, as well as hybrid, custom-built models that build on top of frozen text-image pretrained models. The results show that fine-tuning pretrained models, specifically the FrozenBiLM[229] model, on the LLQA dataset, achieved the best performance while retaining their flexibility and robustness.

Research Question 2 contributes to the advancement of lifelog question answering capabilities by identifying the most suitable QA techniques for lifelog in order to improve the overall performance of lifelog retrieval systems.

Research Question 3 (RQ3). Can incorporating tailored approaches to lifelog question answering result in improved novice user performance on interactive lifelog retrieval tasks, when compared to existing state-of-the-art interactive lifelog retrieval systems?

This research question focuses on the incorporation of tailored question answering approaches into interactive lifelog retrieval tasks by evaluating the system’s performance and comparing it with the existing state-of-the-art lifelog retrieval system (Myscéal). To answer this question, I proposed two sub-questions:

RQ3.1. Does the event-based retrieval support the user to achieve comparative performance to image-based retrieval for lifelog data? To adapt lifelog QA techniques to lifelog retrieval systems, it is necessary for the search results to be in a similar format to the LLQA dataset. In other words, instead of retrieving individual images, the results need to be in the form of ‘events’, which are continuous sequences of lifelog moments (images and their associated metadata). To address this question, I evaluated the performance of the event-based retrieval approach by comparing it with the image-based retrieval approach. The results showed that the event-based retrieval approach can achieve comparative performance to the image-based retrieval approach, while also providing more information to the user. This indicates that the event-based retrieval approach can be used to support lifelog QA tasks.

RQ3.2. Can the tailored question answering approach improve the performance of interactive lifelog retrieval? A pipeline of lifelog question answering was proposed to incorporate lifelog QA techniques into existing lifelog retrieval systems. To demonstrate the effectiveness of the proposed pipeline, a lifelog question answering model was integrated into Myscéal with modifications to develop a dedicated lifelog question answering system called MyEachtra. I reported the results of a user study that evaluated the performance of the tailored question answering approach in interactive lifelog question answering tasks. The results imply that the tailored approach can improve the performance of interactive lifelog retrieval tasks and provide more user satisfaction. Furthermore, MyEachtra’s performance in the LSC’23[71] can be considered state-of-the-art, which further confirms the effectiveness of the proposed pipeline.

Research Question 3 is particularly relevant to the hypothesis as it demonstrates the effectiveness of incorporating lifelog QA techniques into lifelog retrieval systems. The findings of the user studies as well as the benchmarking programme show that the proposed pipeline can improve the performance of interactive lifelog retrieval tasks and extend the capabilities of lifelog retrieval systems to address question answering tasks. Therefore, the findings of this research work support the hypothesis.

8.2 Research Contributions

The main contributions of this thesis can be summarised as follows. The first contribution involves the development of a state-of-the-art interactive lifelog retrieval system, Myscéal which has undergone progressive enhancements to adapt to the evolving landscape of computer vision techniques. Chapter 4 outlined the details of the developed systems and highlights the lessons learned from each research cycle. In the process, a novel ranking algorithm was proposed to support the retrieval of lifelog data, aTF-IDF, which is modified from the traditional TF-IDF algorithm to support images. Moreover, the idea of supporting temporal queries was introduced with Myscéal which is a novel feature that had not been explored in previous lifelog retrieval systems. The results of the user studies and benchmarking programmes showed that the approaches proposed in Myscéal can improve the performance of lifelog retrieval systems. Second, the development of the LLQA dataset, which is the first lifelog question answering dataset, was a significant contribution to the field of lifelog question answering. Chapter 5 described the creation of the LLQA dataset and the comprehensive evaluation of multiple QA techniques on this novel dataset. Another contribution is the analysis of the performance of multiple QA techniques on the LLQA dataset in the benchmarking study. It indicated that fine-tuning pretrained video-language models is a viable approach to lifelog question answering tasks. Third, the first event-based lifelog retrieval system, MyEachtra, was developed to shift the focus from individual images to continuous sequences of lifelog moments (events). Chapter 6 delved into the development of the MyEachtra system and provided a comparison between the event-based and image-based retrieval approaches. The results showed that the event-based retrieval approach can achieve comparative performance to the image-based retrieval approach, while also providing more information to the user and being more suitable for lifelog QA tasks. Lastly, a novel pipeline was introduced to incorporate lifelog QA techniques into lifelog retrieval systems. Chapter 7 presented the pipeline and demonstrated its effectiveness by incorporating lifelog question answering techniques into the MyEachtra system. The results underscore the potential of the pipeline to improve the performance of interactive lifelog QA tasks and provide higher user satisfaction.

8.3 Limitations

Although the proposed approaches have shown promising results, there are still some limitations that were encountered during the research process. In this section, I will discuss some limitations of the proposed approaches and suggest some possible solutions as follows:

- **Strong focus on images, not much on other modalities.** Despite our best efforts to incorporate other modalities to encompass the multimodal nature of lifelog data,

the proposed approaches in this thesis focus mainly on lifelog images with some metadata (for example time and location). This is true not only for the lifelog retrieval systems such as Myscéal and MyEachtra, but also for the lifelog QA dataset, LLQA. Most of the questions in the LLQA dataset are based on visual information as the annotation interface was not efficient to support other modalities. A dedicated effort to design a more comprehensive annotation interface is required to support the annotation of lifelog question answering datasets to increase the diversity of questions. Therefore, LLQA may not be representative of the questions that are asked by real users and could introduce bias to the evaluation of lifelog QA techniques.

- **Question generation is not perfect.** The questions in the LLQA dataset were automatically generated from lifelog captions using a question generation model. Although the automatic question generation process intended to reduce the annotation effort, I found that the review process was time-consuming and required a lot of effort to ensure the quality of the questions as well as the answer candidates. In some cases, the question can be too obvious to answer. In other cases, the candidate answers are too similar to each other, which makes it difficult to choose the correct one. Therefore, the quality of the questions in the LLQA dataset may not be as good as the questions that are manually generated by humans.

- **Small dataset size.** The LLQA dataset is relatively small compared to other datasets in the field of question answering, which made it difficult to train complex models and overfitting was a common problem during the training process. Increasing the size of the LLQA dataset allows the models to learn more complex patterns and be more robust to unseen data. This could be done by using crowdsourcing to annotate the dataset, which is a more cost-effective approach than manual annotation.

- **Small number of users.** The user studies in this thesis were conducted with a relatively small number of users because it is difficult to find users who are willing to participate. This makes it difficult to generalise the results to a wider population. Therefore, the performance of the proposed approaches can be further improved by conducting user studies with a larger number of users. In addition, the user studies are conducted in a controlled environment, which makes it difficult to simulate real-world scenarios where we can identify the users' information needs and learn more about the users' interaction with the system.

- **Limited resources** While most state-of-the-art video QA models rely heavily on pretraining on large-scale datasets such as WebVid[14], the dedicated models for LLQA in this thesis were trained directly on the LLQA dataset due to the limited resources (storage and computing power). Therefore, the designs of the custom-built models are relatively simple, which limits their performance. This is confirmed by the results of the benchmarking study, which showed that finetuning pretrained video-language models on the LLQA dataset achieved the best performance.

8.4 Future Works

As the task of lifelog question answering is very new, there are a considerable number of aspects that can be further explored to improve the performance of lifelog question answering systems. A few suggestions for future works are as follows:

- **Dynamic event segmentation.** This is a challenging task, and there are still many improvements that can be made to the existing approaches. In this thesis, I used a simple approach to dynamic event segmentation, which exploits the visual similarity between images as well as location and time information. However, the definition of an event is not always clear and oftentimes depends on the query and user’s perspective. For instance, considering the query ‘*What is my favourite country for holidays?*’, an event can be as long as several days or even weeks. However, for the query ‘*What restaurant did I go to last night?*’, an event can be as short as a few hours. Therefore, a user-adjustable dynamic event segmentation approach has great potential to support various types of queries and improve the performance of lifelog retrieval systems.
- **Investigating other modalities.** As previously discussed, the methods presented in this thesis primarily focus on images, with limited consideration given to other data modalities like biometrics. To address this limitation, it is worthwhile to conduct benchmarking studies focusing on biometrics data to identify the type of scenarios that require biometrics information and to explore methods for integrating biometric data into lifelog systems. Another direction is to explore the design aspects of a user interface that can effectively visualise, and allow the user to interact with, biometrics data. This allows the user to explore the relationship between biometrics data and other modalities such as images and locations. Such an interface can be used to support various lifelog tasks, including but not limited to annotation, query formulation (for benchmarking purposes), retrieval, and question answering.
- **Improving the lifelog QA pipeline.** The flexible pipeline can be further improved by incorporating additional elements such as *question classification* and *answer postprocessing* modules. The first module can identify the question type, which in turn improves the performance of the Event Retriever component in the lifelog QA pipeline. For example, the question classification module can determine the data modality required to answer the question, for example location, time, visual information, or a combination of these aspects. It can also guide the dynamic event segmentation approach mentioned earlier. The second module, answer postprocessing, takes the answers generated by the lifelog QA model and processes them to improve the quality of the answers. This includes tasks like filtering out answers

that are not relevant to the question type and re-ranking the answers using a separate model.

- **End-to-End Lifelog QA Model.** The pipeline for incorporating lifelog QA techniques into lifelog retrieval systems can be extended to support end-to-end lifelog QA models. In other words, instead of using separate models for the Event Retriever and Event Reader, the model can be trained in an end-to-end fashion, allowing the Event Retriever to learn based on the Event Reader’s output. While inspiration can be drawn from the field of open-domain question answering, it remains unclear whether the end-to-end approach can outperform the pipeline approach, which offers greater flexibility for adapting state-of-the-art models to individual components.
- **Large Language Models (LLMs).** The integration of LLMs into the domain of lifelog question answering represents a promising direction for future research. As of the time of writing, LLMs such as OpenAI’s GPT-3.5 [162] and GPT-4 [2], Facebook’s LLaMA [202], and Google’s Gemini[201] have demonstrated impressive performance in complex decision-making and analytical tasks. The ability to extend these models to lifelog question answering tasks is an exciting work that can be the next step in the development of lifelog retrieval systems! Retrieval Augmented Generation (RAG) [121] is a promising approach that can be used to integrate LLMs into lifelog retrieval systems. RAG is a method that combines the strengths of retrieval-based and generation-based models, resembling the pipeline approach proposed in this thesis. RAG has been expanded into multimodal settings [32], which has enormous potential to support lifelog question answering tasks. This is an exciting direction for future research that can significantly improve the model’s understanding of the user’s information needs and, in turn, improve the quality of the answers.

Appendix A

List of Lifelog Queries Used in Lifelog Experiments

User Study in Chapter 7

In this appendix, I present the list of queries used in the user study in Chapter 7. The queries were chosen randomly from the test collection described in the same chapter. Each user was asked to answer 16 questions, 8 from each system.

Questions for User 1

System	Type	Question	Answer
E-Myscéal	Location	Where was I headed when I took the taxi from the maglev station?	A hotel
E-Myscéal	Object	How did I travel to the Brazen Head where I met my friends?	Taxi
E-Myscéal	Counting	How many salt lamps did I purchase along with a wicker basket in May 2018?	2
E-Myscéal	Time	It was in May 2018. What was the date that I attended a breakfast meeting with other people in a hotel in Dublin?	May 10, 2018
E-Myscéal	Frequency	How often did I have breakfast in a hotel in May 2018?	8 times
E-Myscéal	OCR	What is the maximum number of KGs of clothes that my washing machine at home can take?	10
E-Myscéal	Color	What color was the bottom of the wicker picnic basket that I bought from Carraig Donn?	White

Appendix A. List of Lifelog Queries Used in Lifelog Experiments

E-Myscéal	Duration	How long did it take me to get from home to Clayton Hotel for the breakfast meeting in May 2018?	Half an hour
MyEachtra	Time	In what year did I see the four red alien figures on a painting on a red wall?	2015
MyEachtra	Frequency	How many different shops did I go to on the afternoon of 15/03/2015?	3
MyEachtra	Color	What color was the taxi I initially took from the railway station in China on the 20th of March?	Red
MyEachtra	OCR	What is my car’s registration plate number?	06-D-58377
MyEachtra	Location	What is the name of the American restaurant in Thailand where I had steak?	The Duke’s
MyEachtra	Object	What kind of restaurant was the red building outside of the railway station in China?	Dumpling
MyEachtra	Counting	How many grandfather clocks did I see in the antiques emporium in Sheffield?	3
MyEachtra	Duration	How many minutes did I spend praying in the tunnel with the small golden Buddha?	About 2 minutes

Questions for User 2

System	Type	Question	Answer
E-Myscéal	Duration	How long did it take me to drive to the shopping mall?	An hour
E-Myscéal	Object	How did I get back to my office after Angelina’s Cafe?	Taxi
E-Myscéal	Counting	How many blue paintings were there in the room where a lunchtime ceremony was held back in 2018 in DCU?	2
E-Myscéal	Time	Which month did I visit the museum where I saw two ancient Chinese vases?	May
E-Myscéal	Frequency	How many nights did I drink Greek wine at dinner when I was in Greece in 2019?	2
E-Myscéal	Color	What colour is a 10000 won (Korean currency) bill?	Green
E-Myscéal	Location	Where did I experience the black and white VR roller-coaster game with a handheld controller?	Science Gallery Cafe
E-Myscéal	OCR	What was the name of the airline I took when I flew from Bangkok to Dublin in March 2019?	Turkish Airlines

Appendix A. List of Lifelog Queries Used in Lifelog Experiments

MyEachtra	OCR	I normally wear shirts, but what is the brand of the grey t-shirt that I wore at the start of covid-time?	Abercrombie & Fitch
MyEachtra	Frequency	Which airline did I fly with most often in 2019?	Turkish Airlines
MyEachtra	Duration	How long did I stay in the fast maglev train in China in 2015?	15 minutes
MyEachtra	Time	On which specific date in May 2018 was I looking for my yellow staff card?	May 8, 2018
MyEachtra	Counting	How many salt lamps did I purchase along with a wicker basket in May 2018?	2
MyEachtra	Location	What is the name of the antiques shop did I visit where I photographed the grandfather clocks?	The Antiques Emporium
MyEachtra	Color	What color were the chairs next to the door under the chandeliers in an antique room?	Red
MyEachtra	Object	What is the gender of the person that I had breakfast with at Yeats Country Hotel in Sligo, Ireland?	Female

Questions for User 3

System	Type	Question	Answer
E-Myscéal	Frequency	How many times did I have a BBQ in the garden at home in May 2018?	9
E-Myscéal	Time	What month of the year did I watch the Beatles rooftop concert on TV?	April
E-Myscéal	Counting	How many orange lights on the ceiling in the building where I had a meeting before going to Brown Thomas?	Two
E-Myscéal	Location	I had a TV crew come to my house in 2016. Where did we go after that?	Dublin City University
E-Myscéal	Duration	I was planning a thesis with a PhD student on a whiteboard on a Tuesday in 2016 in my office. How long did it last?	20 minutes
E-Myscéal	OCR	What is the maximum number of KGs of clothes that my washing machine at home can take?	10
E-Myscéal	Color	What color was the taxi I initially took from the railway station in China on the 20th of March?	Red

Appendix A. List of Lifelog Queries Used in Lifelog Experiments

E-Myscéal	Object	Where did I put my yellow staff card on a Tuesday afternoon in May 2018?	Wallet
MyEachtra	OCR	What beverage was I drinking at home during the BBQ in the summer of 2018?	Budweiser beer
MyEachtra	Duration	How long did the flight from Dublin to London take in March 2015?	1.5 hours
MyEachtra	Location	In which city was I when I saw the two vinyl LPs (records) on the table in a hotel in Thailand?	Chiang Mai
MyEachtra	Time	On what date in 2019 did I go homewares shopping around midnight in Ireland?	24/12
MyEachtra	Color	Who color of the tie that the man who was with me in the Chinese museum was wearing?	Red
MyEachtra	Counting	How many blue paintings were there in the room where a lunchtime ceremony was held back in 2018 in DCU?	2
MyEachtra	Frequency	How many times did I have fast food in China in 2018?	2
MyEachtra	Object	What beer did I drink when I had dinner at Asiatique, the outdoor shopping center?	Chang

Questions for User 4

System	Type	Question	Answer
E-Myscéal	Duration	How long did I stop to take a photo of a lake near Sheffield with my Sony camera?	3 minutes
E-Myscéal	Color	What color is my staff card?	Yellow
E-Myscéal	OCR	What is the brand of the car I drive?	Volvo
E-Myscéal	Counting	How many people were with me during my visit to the house with the stone shed/hovel on April 29, 2020?	One
E-Myscéal	Frequency	How many times did I have fast food in China in 2018?	2
E-Myscéal	Object	What religious figure did I see in front of a window in Dublin City University on an August day?	Mother Mary
E-Myscéal	Location	What is the name of the building where I saw two large blue paintings of the sea with islands and sky?	The Helix
E-Myscéal	Time	On what date did I change my office in 2020?	09/03
MyEachtra	Frequency	How many times did I shave in September 2016?	2

Appendix A. List of Lifelog Queries Used in Lifelog Experiments

MyEachtra	Color	What is the color of the jacket worn by the black and white panda-bear toy that can sometimes be seen with the two long rabbits?	Blue
MyEachtra	Duration	How long did I play the VR rollercoaster game at Science Gallery Cafe?	3 minutes
MyEachtra	OCR	What airline company operated the last flight I had in May 2018?	Turkish Air-line
MyEachtra	Object	How did I travel to the Brazen Head where I met my friends?	Taxi
MyEachtra	Time	What month did the conference MMM happen in 2019?	January
MyEachtra	Location	Which country was I heading to when I took the photograph of an A380 airplane in Germany?	China
MyEachtra	Counting	How many white cars were there at the Red House in Howth on a beautiful blue-sky day?	One

Questions for User 5

System	Type	Question	Answer
E-Myscéal	Time	In which month and year did I get my car's wheel repaired?	March 2015
E-Myscéal	Color	What color was the building that was next to the Red House in Howth?	White
E-Myscéal	Location	Which shop did I buy some salt lamps and a wicker basket from in the summer of 2018?	Carraig Donn
E-Myscéal	Frequency	How many times have I visited the house with the stone shed/hovel in April 2020?	2
E-Myscéal	OCR	What is my car's registration plate number?	06-D-58377
E-Myscéal	Counting	How many lotus vases were around the small golden Buddha in the tunnel in Thailand?	2
E-Myscéal	Object	What is the gender of the person that I had breakfast with at Yeats Country Hotel in Sligo, Ireland?	Female
E-Myscéal	Duration	How long did I wait in the queue to order the food at McDonald's in an airport in China?	2 minutes
MyEachtra	OCR	What platform was I using to watch the Beatles concert on TV?	YouTube

Appendix A. List of Lifelog Queries Used in Lifelog Experiments

MyEachtra	Location	What was the name of the shop where I bought hand soaps in the early morning?	Molton Brown
MyEachtra	Color	What is the color of the “For Sale” sign in front of the stone cottage I saw in April?	Blue
MyEachtra	Counting	How many blue paintings were there in the room where a lunchtime ceremony was held back in 2018 in DCU?	2
MyEachtra	Object	Where in the house did I feed my friend’s dog when he kept asking me for food in May 2018?	Garden
MyEachtra	Time	When did I eat out at Sole restaurant in 2018?	May 31, 2018
MyEachtra	Frequency	How many nights did I drink Greek wine at dinner when I was in Greece in 2019?	2
MyEachtra	Duration	How long did my meeting at Angelina’s Cafe last?	2 minutes

Questions for User 6

System	Type	Question	Answer
E-Myscéal	Color	What is the colour of the jacket worn by the black and white panda-bear toy that can sometimes be seen with the two long rabbits?	Blue
E-Myscéal	OCR	I normally wear shirts, but what is the brand of the grey t-shirt that I wore at the start of covid-time?	Abercrombie & Fitch
E-Myscéal	Object	What kind of material was the picnic basket that I bought from Carraig Donn made of?	Wicker
E-Myscéal	Duration	How long did it take for the man at the garage to fix my old car’s wheel?	10 minutes
E-Myscéal	Frequency	How many different shops did I go to on the afternoon of 15/03/2015?	3
E-Myscéal	Time	When did I put a ‘no junk mail’ sign on my door?	March 19, 2015
E-Myscéal	Location	Where did I go to get new keys in 2015?	Northside Shopping Centre
E-Myscéal	Counting	How many people did I have a BBQ with on the 14th of May in 2018?	4
MyEachtra	Color	I was following Rami from Angelina’s Cafe. What color was his backpack?	Red

Appendix A. List of Lifelog Queries Used in Lifelog Experiments

MyEachtra	Location	What coffee shop did I go to after seeing a Mother Mary poster at Dublin City University?	The Devlin Hotel
MyEachtra	Duration	How long did I stop driving to take photos of a lake in the United Kingdom with my Sony camera?	3 minutes
MyEachtra	Time	When did I visit the antique store and take photos of the grandfather clocks?	March 7, 2015
MyEachtra	Object	It was on a Thursday in May. What breakfast food did I have at Clayton Hotel in Dublin?	Croissant
MyEachtra	Counting	How many nights did I stay at Yeats Country Hotel in Sligo, Ireland?	1
MyEachtra	Frequency	How many times did I shave in September 2016? (recount the times)	2
MyEachtra	OCR	What was the number of my office door (in 2019)? It was on the second floor, at Dublin City University.	L2.42

Questions for User 7

System	Type	Question	Answer
E-Myscéal	Time	Which month of the year did I go to a tourist park and see a large ornamental tower with lions (maybe) on top?	October
E-Myscéal	Counting	How many orange lights were on the ceiling in the building where I had a meeting before going to Brown Thomas?	Two
E-Myscéal	Color	What color is my staff card?	Yellow
E-Myscéal	Frequency	Which airline did I fly with most often in 2019?	Turkish Airlines
E-Myscéal	OCR	What airline company operated the last flight I had in May 2018?	Turkish Airline
E-Myscéal	Location	After meeting at the Brazen Head, where did my friends and I go for drinks?	Temple Bar
E-Myscéal	Object	What was the weather like when I went for a short walk in Wicklow in 2019?	Cold
E-Myscéal	Duration	How many days did it take for me to build my computer back in March 2015?	2
MyEachtra	Time	On which specific date in May 2018 was I in China and drinking a ‘Corona Extra’ beer?	May 23, 2018

Appendix A. List of Lifelog Queries Used in Lifelog Experiments

MyEachtra	Location	Where did I go to before going to Angelina’s Cafe in May?	AIB ATM
MyEachtra	Object	Which wheel of my car needed repair when I brought it into a car repair shop?	Front Left
MyEachtra	Duration	How long did I play the VR rollercoaster game at Science Gallery Cafe?	3 minutes
MyEachtra	Color	What color is a 10000 won (Korean currency) bill?	Green
MyEachtra	Counting	How many people came to my home for a TV recording?	3
MyEachtra	Frequency	How often did I have breakfast in a hotel in May 2018?	8 times
MyEachtra	OCR	What was written on the blue Mother Mary poster I saw in All Hallows Campus?	Pray For Us

Questions for User 8

System	Type	Question	Answer
E-Myscéal	Location	It was an afternoon in the middle of March 2015. What is the name of the shop where I saw a T-shirt for sale that says “I love bicycle”?	Halfords
E-Myscéal	OCR	What song from Dire Straits was written on the diamond-shaped wooden sign by the sea?	Telegraph Road
E-Myscéal	Duration	I bought some hand soaps in a shop in the early morning. It was in an outdoor shopping mall. How long did it take me to drive E-Myscéal	An hour
E-Myscéal	Color	What color was the staff uniform in Yeats Country Hotel?	White
E-Myscéal	Time	What month of the year did I watch the Beatles rooftop concert on TV?	April
E-Myscéal	Object	Where did I put my yellow staff card on a Tuesday afternoon in May 2018?	Wallet
E-Myscéal	Frequency	How many nights did I drink Greek wine at dinner when I was in Greece in 2019?	2
E-Myscéal	Counting	How many grandfather clocks did I see in the antiques emporium in Sheffield?	3
MyEachtra	Counting	How many people were with me during my visit to the house with the stone shed/hovel on April 29, 2020?	One

Appendix A. List of Lifelog Queries Used in Lifelog Experiments

MyEachtra	Object	What flavor of ice cream did I have by the sea before seeing the Red House?	Vanilla
MyEachtra	Location	What store did I go to shop for blue cups on a Wednesday evening?	Donaghmede Shopping Centre
MyEachtra	Time	On which specific date in May 2018 was I in China and drinking a ‘Corona Extra’ beer?	May 23, 2018
MyEachtra	Duration	How long did it take me to walk to Dublin Pearse Railway Station after the sushi bar?	36 minutes
MyEachtra	Color	What was the color of the tie that the man who was with me in the Chinese museum was wearing?	Red
MyEachtra	OCR	What airline did I fly on for my first flight in 2020? I remember it was a small plane, perhaps an ATR-72.	Stobart Air
MyEachtra	Frequency	How many different airports did I go to in May 2019?	5

Questions for User 9

System	Type	Question	Answer
E-Myscéal	Color	What colour was the taxi sign of the taxi I took to Daejeon station in Korea?	Purple
E-Myscéal	Object	What type of dog does my sister have?	Golden Retriever
E-Myscéal	Duration	How long did my meeting at Angelina’s Cafe last?	Half an hour
E-Myscéal	Frequency	How many times did I have a BBQ in the garden at home in May 2018?	9
E-Myscéal	Counting	How many orange lights were on the ceiling in the building where I had a meeting before going to Brown Thomas?	Two
E-Myscéal	Location	What is the name of the antique shop I visited where I photographed the grandfather clocks?	The Antiques Emporium
E-Myscéal	OCR	What airline was operating the A380 airplane that I photographed in Germany?	Lufthansa
E-Myscéal	Time	What date did a TV crew come to my home to record some video?	Sep 21, 2019

Appendix A. List of Lifelog Queries Used in Lifelog Experiments

MyEachtra	Color	What were the three colors of the pens we used while planning the thesis on the whiteboard in 2016?	Black, white, blue
MyEachtra	OCR	What is the brand of car I drive?	Volvo
MyEachtra	Object	What religious figure did I see in front of a window in Dublin City University on an August day?	Mother Mary
MyEachtra	Location	What was the name of the shop where I bought hand soaps in the early morning?	Molton Brown
MyEachtra	Counting	How many people were with me during my visit to the house with the stone shed/hovel on April 29, 2020?	One
MyEachtra	Duration	How long did I wait in the queue to order the food at McDonald's in an airport in China?	2 minutes
MyEachtra	Frequency	How often did I visit religious sites in September 2019?	never
MyEachtra	Time	On which specific date in May 2018 was I in China and drinking a 'Corona Extra' beer?	May 23, 2018

Questions for User 10

System	Type	Question	Answer
E-Myscéal	OCR	What airline did I fly on for my first flight in 2020? I remember it was a small plane, perhaps an ATR-72.	Stobart Air
E-Myscéal	Location	What is the name of the car shop I went to after seeing the Blue Air aircraft back in 2018?	Joe Duffy
E-Myscéal	Counting	How many people were with me during my visit to the house with the stone shed/hovel on April 29, 2020?	One
E-Myscéal	Color	What color was the taxi I initially took from the railway station in China on the 20th of March?	Blue
E-Myscéal	Object	What did I get right after buying hand soaps in Molton Brown?	Coffee
E-Myscéal	Time	Which year's summer did I have 9 BBQs in a month?	2015
E-Myscéal	Frequency	Which airline did I fly with most often in 2019?	Turkish Airlines
E-Myscéal	Duration	How long did it take for the man at the garage to fix my old car's wheel?	40 minutes
MyEachtra	Time	What year did I visit Northside Shopping Centre to get new keys?	2018

Appendix A. List of Lifelog Queries Used in Lifelog Experiments

MyEachtra	Color	What is the colour of the jacket worn by the black and white panda-bear toy that can sometimes be seen with the two long rabbits?	Red
MyEachtra	Counting	How many orange lights were on the ceiling in the building where I had a meeting before going to Brown Thomas?	One
MyEachtra	OCR	What was the brand of the camera I used to take a photo of the lake in the United Kingdom in 2015?	Sony
MyEachtra	Duration	How long did I stay in Daejeon Station?	10 minutes
MyEachtra	Location	What is the name of the shopping centre did I visit after checking out of the hotel in Sligo in 2016?	Quayside
MyEachtra	Object	What kind of alcohol did I consider buying for emergency food supplies during my visit to Musgrave MarketPlace in February?	Whiskey
MyEachtra	Frequency	How often did I go running in the park near my home on Saturday mornings in February?	One time

User Experience Questionnaire in Chapter 7

The user experience questionnaire used in the user study in Chapter 7 is presented in this appendix. The questionnaire was used to measure the user experience of the participants after they interacted with the two lifelog retrieval systems. The questionnaire was adapted from the User Experience Questionnaire (UEQ) [109].

Instructions: Please rate your experience with the system you have just used. The scale ranges from -3 to 3, where -3 means very bad, 0 means neutral, and 3 means very good.

Table A.11: User Experience Questionnaire

obstructive	-3	-2	-1	0	1	2	3	supportive
complicated	-3	-2	-1	0	1	2	3	easy
inefficient	-3	-2	-1	0	1	2	3	efficient
confusing	-3	-2	-1	0	1	2	3	clear
boring	-3	-2	-1	0	1	2	3	exciting
not interesting	-3	-2	-1	0	1	2	3	interesting
conventional	-3	-2	-1	0	1	2	3	inventive
usual	-3	-2	-1	0	1	2	3	leading edge

References

- [1] Fatma Abdallah, Ghada Feki, Ben ammar Anis, and Chokri Ben Amar. *Big Data For Lifelog Moments Retrieval Improvement*. July 23, 2019.
- [2] Josh Achiam et al. “Gpt-4 technical report”. In: *ArXiv preprint abs/2303.08774* (2023).
- [3] Naushad Alam, Yvette Graham, and Cathal Gurrin. “Memento: A Prototype Lifelog Search Engine for LSC’21”. In: *Proceedings of the 4th Annual on Lifelog Search Challenge*. 2021, pp. 53–58.
- [4] Naushad Alam, Yvette Graham, and Cathal Gurrin. “Memento 2.0: An Improved Lifelog Search Engine for LSC’22”. In: *Proceedings of the 5th Annual on Lifelog Search Challenge*. 2022, pp. 2–7.
- [5] Naushad Alam, Yvette Graham, and Cathal Gurrin. “Memento 3.0: An Enhanced Lifelog Search Engine for LSC’23”. In: *Proceedings of the 6th Annual ACM Lifelog Search Challenge*. 2023, pp. 41–46.
- [6] Ahmed Alateeq, Mark Roantree, and Cathal Gurrin. “Voxento 2.0: a prototype voice-controlled interactive search engine for lifelogs”. In: *Proceedings of the 4th Annual on Lifelog Search Challenge*. 2021, pp. 65–70.
- [7] Ahmed Alateeq, Mark Roantree, and Cathal Gurrin. “Voxento 4.0: A More Flexible Visualisation and Control for Lifelogs”. In: *Proceedings of the 6th Annual ACM Lifelog Search Challenge*. 2023, pp. 7–12.
- [8] Adrià Alsina, Xavier Giró, and Cathal Gurrin. “An interactive lifelog search engine for lsc2018”. In: *Proceedings of the 2018 ACM Workshop on The Lifelog Search Challenge*. 2018, pp. 30–32.
- [9] Peter Anderson et al. “Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering”. In: *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. IEEE Computer Society, 2018, pp. 6077–6086.

-
- [10] Mihael Ankerst, Markus M Breunig, Hans-Peter Kriegel, and Jörg Sander. “OPTICS: Ordering points to identify the clustering structure”. In: *ACM Sigmod record* 28.2 (1999), pp. 49–60.
- [11] Stanislaw Antol et al. “VQA: Visual Question Answering”. In: *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. IEEE Computer Society, 2015, pp. 2425–2433.
- [12] Rama Ahmadi Ariayudha, Hilal H. Nuha, and Muhammad Irsan. “Reading Stress Levels and Setting Emotional Patterns with Sensors-based on The Galvanic Skin Response (GSR) Method”. In: *2023 International Conference on Data Science and Its Applications (ICoDSA)*. IEEE, 2023.
- [13] Aleks Aris, Jim Gemmell, and Roger Lueder. “Exploiting location and time for photo search and storytelling in MyLifeBits”. In: (2004).
- [14] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. “Frozen in Time: A Joint Video and Image Encoder for End-to-End Retrieval”. In: *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*. IEEE, 2021, pp. 1708–1718.
- [15] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. “A CLIP-Hitchhiker’s Guide to Long Video Retrieval”. In: *ArXiv preprint abs/2205.08508* (2022).
- [16] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. “BEiT: BERT Pre-Training of Image Transformers”. In: *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.
- [17] Hangbo Bao et al. “UniLMv2: Pseudo-Masked Language Models for Unified Language Model Pre-Training”. In: *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*. Vol. 119. Proceedings of Machine Learning Research. PMLR, 2020, pp. 642–652.
- [18] Razieh Baradaran, Razieh Ghiasi, and Hossein Amirkhani. “A Survey on Machine Reading Comprehension Systems”. In: *Natural Language Engineering* 28.6 (2022), pp. 683–732.
- [19] Philip J Barnard, Fionnuala C Murphy, Maria Teresa Carthery-Goulart, Cristina Ramponi, and Linda Clare. “Exploring the basis and boundary conditions of SenseCam-facilitated recollection”. In: *Memory* 19.7 (2011), pp. 758–767.
- [20] Anahid Basiri et al. “Indoor location based services challenges, requirements and usability of current solutions”. In: *Computer Science Review* 24 (2017), pp. 1–12.
- [21] Emma Berry et al. “The use of a wearable camera, SenseCam, as a pictorial diary to improve autobiographical memory in a patient with limbic encephalitis: A preliminary report”. In: *Neuropsychological rehabilitation* 17.4-5 (2007), pp. 582–601.
-

-
- [22] Tom B. Brown et al. “Language Models are Few-Shot Learners”. In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Ed. by Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin. 2020.
- [23] Paul Bruton, Joss Langford, Matthew Reed, and David Snelling. *Classification of Everyday Living Version 1.0*. 2019.
- [24] Vannevar Bush et al. “As we may think”. In: *The atlantic monthly* 176.1 (1945), pp. 101–108.
- [25] Daragh Byrne, Aisling Kelliher, and Gareth J. F. Jones. “Life editing: third-party perspectives on lifelog content”. In: *Proceedings of the International Conference on Human Factors in Computing Systems, CHI 2011, Vancouver, BC, Canada, May 7-12, 2011*. Ed. by Desney S. Tan, Saleema Amershi, Bo Begole, Wendy A. Kellogg, and Manas Tungare. ACM, 2011, pp. 1501–1510.
- [26] Santiago Castro et al. “LifeQA: A Real-life Dataset for Video Question Answering”. English. In: *Proceedings of the Twelfth Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, 2020, pp. 4352–4358.
- [27] Chia-Chun Chang, Min-Huan Fu, Hen-Hsen Huang, and Hsin-Hsi Chen. “An interactive approach to integrating external textual knowledge for multimodal lifelog retrieval”. In: *Proceedings of the ACM workshop on lifelog search challenge*. 2019, pp. 41–44.
- [28] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. “Conceptual 12M: Pushing Web-Scale Image-Text Pre-Training To Recognize Long-Tail Visual Concepts”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, 2021, pp. 3558–3568.
- [29] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. “Reading Wikipedia to Answer Open-Domain Questions”. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, 2017, pp. 1870–1879.
- [30] Kan Chen, Jiang Wang, Liang-Chieh Chen, Haoyuan Gao, Wei Xu, and Ram Nevatia. “Abc-cnn: An attention based convolutional neural network for visual question answering”. In: *ArXiv preprint abs/1511.05960* (2015).
-

-
- [31] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. “Encoder-decoder with atrous separable convolution for semantic image segmentation”. In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 801–818.
- [32] Wenhui Chen, Hexiang Hu, Xi Chen, Pat Verga, and William Cohen. “MuRAG: Multimodal Retrieval-Augmented Generator for Open Question Answering over Images and Text”. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, 2022, pp. 5558–5570.
- [33] Duc-Tien Dang-Nguyen, Luca Piras, Michael Riegler, Giulia Boato, Liting Zhou, and Cathal Gurrin. “Overview of ImageCLEFlifelog 2017: Lifelog Retrieval and Summarization”. In: *CLEF (Working Notes)*. 2019, p. 23.
- [34] Duc-Tien Dang-Nguyen, Luca Piras, Michael Riegler, Liting Zhou, Mathias Lux, and Cathal Gurrin. “Overview of ImageCLEFlifelog 2018: Daily Living Understanding and Lifelog Moment Retrieval”. In: *CLEF (Working Notes)*. 2018, p. 19.
- [35] Duc-Tien Dang-Nguyen et al. “Overview of ImageCLEFlifelog 2019: Solve My Life Puzzle and Lifelog Moment Retrieval”. In: *CLEF (Working Notes)*. 2019, p. 17.
- [36] Minh-Son Dao, Anh-Khoa Vo, Trong-Dat Phan, and K. Zettsu. “BIDALImageCLEFlifelog2019: The Role of Content and Context of Daily Activities in Insights from Lifelogs”. In: *Clef*. 2019.
- [37] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. “ImageNet: A large-scale hierarchical image database”. In: *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*. IEEE Computer Society, 2009, pp. 248–255.
- [38] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, 2019, pp. 4171–4186.
- [39] Ming Ding, Chang Zhou, Qibin Chen, Hongxia Yang, and Jie Tang. “Cognitive Graph for Multi-Hop Reading Comprehension at Scale”. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, 2019, pp. 2694–2703.
- [40] Mihai Dogariu and Bogdan Ionescu. “Multimedia Lab ImageCLEF 2018 Lifelog Moment Retrieval Task”. In: *CLEF (Working Notes)*. 2018, p. 13.
-

-
- [41] Aiden R Doherty, Alan F Smeaton, Keansub Lee, and Daniel PW Ellis. “Multi-modal segmentation of lifelog data”. In: (2007).
- [42] Aiden R Doherty et al. “Passively recognising human activities through lifelogging”. In: *Computers in human behavior* 27.5 (2011), pp. 1948–1958.
- [43] Aiden R Doherty et al. “Experiences of aiding autobiographical memory using the SenseCam”. In: *Human–Computer Interaction* 27.1-2 (2012), pp. 151–174.
- [44] Jianfeng Dong, Xirong Li, and Cees GM Snoek. “Predicting visual features from text for image and video caption retrieval”. In: *IEEE Transactions on Multimedia* 20.12 (2018), pp. 3377–3388.
- [45] Li Dong et al. “Unified Language Model Pre-training for Natural Language Understanding and Generation”. In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*. Ed. by Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d’Alché-Buc, Emily B. Fox, and Roman Garnett. 2019, pp. 13042–13054.
- [46] Aaron Duane, Cathal Gurrin, and Wolfgang Huerst. “Virtual Reality Lifelog Explorer: Lifelog Search Challenge at ACM ICMR 2018”. In: *Proceedings of the 2018 ACM Workshop on The Lifelog Search Challenge*. Lsc ’18. New York, NY, USA: Association for Computing Machinery, 2018, pp. 20–23.
- [47] Aaron Duane, Bjorn Por Jonsson, and Cathal Gurrin. “VRLE: Lifelog Interaction Prototype in Virtual Reality: Lifelog Search Challenge at ACM ICMR 2020”. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*. Lsc ’20. Dublin, Ireland: Association for Computing Machinery, 2020, pp. 7–12.
- [48] Aaron Duane and Bjorn Þór Jónsson. “ViRMA: Virtual reality multimedia analytics at LSC 2021”. In: *Proceedings of the 4th Annual on Lifelog Search Challenge*. 2021, pp. 29–34.
- [49] Bethan Everson, Kelly A Mackintosh, Melitta A McNarry, Charlotte Todd, and Gareth Stratton. “Can wearable cameras be used to validate school-aged children’s lifestyle behaviours?” In: *Children* 6.2 (2019), p. 20.
- [50] C. Fan. “EgoVQA - An Egocentric Video Question Answering Benchmark Dataset”. In: *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. 2019, pp. 4359–4366.
- [51] Chenyou Fan, Xiaofan Zhang, Shu Zhang, Wensheng Wang, Chi Zhang, and Heng Huang. “Heterogeneous Memory Enhanced Multimodal Attention Model for Video Question Answering”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 1999–2007.
-

- [52] Min-Huan Fu, Chia-Chun Chang, Hen-Hsen Huang, and Hsin-Hsi Chen. “Incorporating external textual knowledge for life event recognition and retrieval”. In: *Proceedings of the 14th NTCIR Conference on Evaluation of Information Access Technologies*. 2019, pp. 61–71.
- [53] Tsu-Jui Fu et al. “An empirical study of end-to-end video-language transformers with masked visual modeling”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 22898–22909.
- [54] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. “Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding”. In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas: Association for Computational Linguistics, 2016, pp. 457–468.
- [55] Haoyuan Gao, Junhua Mao, Jie Zhou, Zhiheng Huang, Lei Wang, and Wei Xu. “Are You Talking to a Machine? Dataset and Methods for Multilingual Image Question”. In: *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*. Ed. by Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett. 2015, pp. 2296–2304.
- [56] Jiyang Gao, Runzhou Ge, Kan Chen, and Ram Nevatia. “Motion-Appearance Co-Memory Networks for Video Question Answering”. In: *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. IEEE Computer Society, 2018, pp. 6576–6585.
- [57] Yifan Gao, Lidong Bing, Piji Li, Irwin King, and Michael R. Lyu. “Generating Distractors for Reading Comprehension Questions from Real Examinations”. In: *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press, 2019, pp. 6423–6430.
- [58] Ralph Gasser, Luca Rossetto, Silvan Heller, and Heiko Schuldt. “Cottontail DB: An Open Source Database System for Multimedia Retrieval and Analysis”. In: *MM ’20: The 28th ACM International Conference on Multimedia, Virtual Event / Seattle, WA, USA, October 12-16, 2020*. 2020, pp. 4465–4468.
- [59] Jim Gemmell, Gordon Bell, Roger Lueder, Steven Drucker, and Curtis Wong. “MyLifeBits: fulfilling the Memex vision”. In: *Proceedings of the tenth ACM international conference on Multimedia*. 2002, pp. 235–238.

-
- [60] Hana Gharbi, Sahbi Bahroun, and Ezzeddine Zagrouba. “A Novel Key Frame Extraction Approach for Video Summarization”. In: *Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*. SCITEPRESS - Science and Technology Publications, 2016.
- [61] Rashmi Gupta. “Considering Documents in Lifelog Information Retrieval”. In: *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*. ICMR '18. ACM, 2018.
- [62] Rashmi Gupta. “Event based information retrieval from digital lifelogs”. PhD thesis. Dublin City University, 2020.
- [63] Cathal Gurrin, Hideo Joho, and Frank Hopfgartner. “Overview of NTCIR-12 Lifelog Task”. In: *12th NTCIR Conference on Evaluation of Information Access Technologies*. 2016, p. 7.
- [64] Cathal Gurrin, Alan F Smeaton, Aiden R Doherty, et al. “Lifelogging: Personal big data”. In: *Foundations and Trends® in information retrieval* 8.1 (2014), pp. 1–125.
- [65] Cathal Gurrin et al. “The smartphone as a platform for wearable cameras in health research”. In: *American journal of preventive medicine* 44.3 (2013), pp. 308–313.
- [66] Cathal Gurrin et al. “A test collection for interactive lifelog retrieval”. In: *Multimedia Modeling: 25th International Conference, MMM 2019, Thessaloniki, Greece, January 8–11, 2019, Proceedings, Part I* 25. Springer. 2019, pp. 312–324.
- [67] Cathal Gurrin et al. “Advances in lifelog data organisation and retrieval at the NTCIR-14 Lifelog-3 task”. In: *NII Testbeds and Community for Information Access Research: 14th International Conference, NTCIR 2019, Tokyo, Japan, June 10–13, 2019, Revised Selected Papers 14*. Springer. 2019, pp. 16–28.
- [68] Cathal Gurrin et al. “Introduction to the Third Annual Lifelog Search Challenge (LSC'20)”. In: *Proceedings of the 2020 International Conference on Multimedia Retrieval*. Icmr '20. New York, NY, USA: Association for Computing Machinery, June 8, 2020, pp. 584–585.
- [69] Cathal Gurrin et al. “Introduction to the Fourth Annual Lifelog Search Challenge, LSC'21”. In: *Proc. International Conference on Multimedia Retrieval (ICMR'21)*. Taipei, Taiwan: Acm, 2021.
- [70] Cathal Gurrin et al. “Introduction to the Fifth Annual Lifelog Search Challenge, LSC'22”. In: *Proc. International Conference on Multimedia Retrieval (ICMR'22)*. Icmr '22. Newark, NJ, USA, 2022.
- [71] Cathal Gurrin et al. “Introduction to the Sixth Annual Lifelog Search Challenge, LSC'23”. In: *Proc. International Conference on Multimedia Retrieval (ICMR'23)*. Icmr '23. Thessaloniki, Greece, 2023.
-

-
- [72] Morgan Harvey, Marc Langheinrich, and Geoff Ward. “Remembering through lifelogging: A survey of human memory augmentation”. In: *Pervasive and Mobile Computing* 27 (2016), pp. 14–26.
- [73] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep Residual Learning for Image Recognition”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 770–778.
- [74] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. “DeBERTa: decoding-Enhanced Bert with Disentangled Attention”. In: *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.
- [75] Silvan Heller, Mahnaz Amiri Parian, Ralph Gasser, Loris Sauter, and Heiko Schuldt. “Interactive Lifelog Retrieval with Vitriivr”. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*. Lsc ’20. Dublin, Ireland: Association for Computing Machinery, 2020, pp. 1–6.
- [76] Silvan Heller et al. “Interactive video retrieval evaluation at a distance: comparing sixteen interactive video search systems in a remote setting at the 10th Video Browser Showdown”. In: *International Journal of Multimedia Information Retrieval* 11.1 (2022), pp. 1–18.
- [77] Karl Moritz Hermann et al. “Teaching Machines to Read and Comprehend”. In: *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*. Ed. by Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett. 2015, pp. 1693–1701.
- [78] Karl Moritz Hermann et al. “Teaching Machines to Read and Comprehend”. In: *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*. Ed. by Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett. 2015, pp. 1693–1701.
- [79] Hevner, March, Park, and Ram. “Design Science in Information Systems Research”. In: *MIS Quarterly* 28.1 (2004), p. 75.
- [80] Nhat Hoang-Xuan et al. “Flexible interactive retrieval SysTem 3.0 for visual lifelog exploration at LSC 2022”. In: *Proceedings of the 5th Annual on Lifelog Search Challenge*. 2022, pp. 20–26.
- [81] Steve Hodges, Emma Berry, and Ken Wood. “SenseCam: A wearable camera that stimulates and rehabilitates autobiographical memory”. In: *Memory* 19.7 (2011), pp. 685–696.
-

-
- [82] Steve Hodges et al. “SenseCam: A retrospective memory aid”. In: *UbiComp 2006: Ubiquitous Computing: 8th International Conference, UbiComp 2006 Orange County, CA, USA, September 17-21, 2006 Proceedings 8*. Springer, 2006, pp. 177–193.
- [83] Maria Tysse Hordvik, Julie Sophie Teilstad Østby, Manoj Kesavulu, Thao-Nhu Nguyen, Tu-Khiem Le, and Duc-Tien Dang-Nguyen. “LifeLens: Transforming Lifelog Search with Innovative UX/UI Design”. In: *Proceedings of the 6th Annual ACM Lifelog Search Challenge*. 2023, pp. 1–6.
- [84] Vlad Hosu, Hanhe Lin, Tamas Sziranyi, and Dietmar Saupe. “KonIQ-10k: An ecologically valid database for deep learning of blind image quality assessment”. In: *IEEE Transactions on Image Processing* 29 (2020), pp. 4041–4056.
- [85] Hsin-Yuan Huang, Chenguang Zhu, Yelong Shen, and Weizhu Chen. “FusionNet: Fusing via Fully-aware Attention with Application to Machine Comprehension”. In: *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. Open-Review.net, 2018.
- [86] Gautier Izacard and Edouard Grave. “Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering”. In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Online: Association for Computational Linguistics, 2021, pp. 874–880.
- [87] Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. “TGIF-QA: Toward Spatio-Temporal Reasoning in Visual Question Answering”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, 2017, pp. 1359–1367.
- [88] Ziwei Ji et al. “Survey of hallucination in natural language generation”. In: *ACM Computing Surveys* 55.12 (2023), pp. 1–38.
- [89] Yangqing Jia et al. “Caffe: Convolutional Architecture for Fast Feature Embedding”. In: *arXiv:1408.5093 [cs]* (June 20, 2014).
- [90] Jeff Johnson, Matthijs Douze, and Hervé Jégou. “Billion-scale similarity search with GPUs”. In: *IEEE Transactions on Big Data* 7.3 (2019), pp. 535–547.
- [91] Dan Jurafsky and James H. Martin. “Chapter 14: Question Answering and Information Retrieval”. In: *Speech and Language Processing*. Ed. by Dan Jurafsky and James H. Martin. Stanford University Press, 2023.
- [92] Kushal Kafle and Christopher Kanan. “Visual question answering: Datasets, algorithms, and future challenges”. In: *Computer Vision and Image Understanding* 163 (2017), pp. 3–20.
-

-
- [93] Vaiva Kalnikaitė, Abigail Sellen, Steve Whittaker, and David S. Kirk. “Now let me see where i was: understanding how lifelogs mediate memory”. In: *Proceedings of the 28th International Conference on Human Factors in Computing Systems, CHI 2010, Atlanta, Georgia, USA, April 10-15, 2010*. Ed. by Elizabeth D. Mynatt, Don Schoner, Geraldine Fitzpatrick, Scott E. Hudson, W. Keith Edwards, and Tom Rodden. ACM, 2010, pp. 2045–2054.
- [94] Vladimir Karpukhin et al. “Dense Passage Retrieval for Open-Domain Question Answering”. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, 2020, pp. 6769–6781.
- [95] Ergina Kavallieratou, Carlos R del-Blanco, Carlos Cuevas, and Narciso García. “Retrieving Events in Life Logging.” In: *CLEF (Working Notes)*. 2018.
- [96] Isadora Nguyen Van Khan, Pranita Shrestha, Min Zhang, Yiqun Liu, and Shaoping Ma. “A two-level lifelog search engine at the lsc 2019”. In: *Proceedings of the ACM Workshop on Lifelog Search Challenge*. 2019, pp. 19–23.
- [97] Omar Shahbaz Khan, Aaron Duane, Björn Þór Jónsson, Jan Zahálka, Stevan Rudinac, and Marcel Worring. “Exquisitor at the Lifelog Search Challenge 2021: Relationships Between Semantic Classifiers”. In: *Proceedings of the 4th Annual on Lifelog Search Challenge*. Icmr ’21. Acm, 2021.
- [98] Omar Shahbaz Khan, Björn Þór Jónsson, Jan Zahálka, Stevan Rudinac, and Marcel Worring. “Exquisitor at the lifelog search challenge 2019”. In: *Proceedings of the ACM Workshop on Lifelog Search Challenge*. 2019, pp. 7–11.
- [99] Omar Shahbaz Khan et al. “Exquisitor at the Lifelog Search Challenge 2020”. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*. Lsc ’20. Dublin, Ireland: Association for Computing Machinery, 2020, pp. 19–22.
- [100] Jin-Hwa Kim et al. “Multimodal Residual Learning for Visual QA”. In: *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*. Ed. by Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett. 2016, pp. 361–369.
- [101] Minkyung Kim, Dong-Wook Lee, Kangseok Kim, Jai-Hoon Kim, and We-Duke Cho. “Predicting personal information behaviors with lifelog data”. In: *9th International Conference and Expo on Emerging Technologies for a Smarter World (CEWIT)*. IEEE, 2012.
-

-
- [102] Emil Knudsen, Thomas Holstein Qvortrup, Omar Shahbaz Khan, and Björn Þór Jónsson. “XQC at the lifelog search challenge 2021: Interactive learning on a mobile device”. In: *Proceedings of the 4th Annual on Lifelog Search Challenge*. 2021, pp. 89–93.
- [103] Oleksandr Kolomiyets and Marie-Francine Moens. “A survey on question answering technology from an information retrieval perspective”. In: *Information Sciences* 181.24 (2011), pp. 5412–5434.
- [104] Ranjay Krishna et al. “Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations”. In: *International Journal of Computer Vision* 123.1 (2017), pp. 32–73.
- [105] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*. Ed. by Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, Léon Bottou, and Kilian Q. Weinberger. 2012, pp. 1106–1114.
- [106] Alina Kuznetsova et al. “The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale”. In: *International Journal of Computer Vision* 128.7 (2020), pp. 1956–1981.
- [107] Tom Kwiatkowski et al. “Natural Questions: A Benchmark for Question Answering Research”. In: *Transactions of the Association for Computational Linguistics* 7 (2019), pp. 452–466.
- [108] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. “RACE: Large-scale ReAding Comprehension Dataset From Examinations”. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, 2017, pp. 785–794.
- [109] Bettina Laugwitz, Theo Held, and Martin Schrepp. “Construction and evaluation of a user experience questionnaire”. In: *HCI and Usability for Education and Work: 4th Symposium of the Workgroup Human-Computer Interaction and Usability Engineering of the Austrian Computer Society, USAB 2008, Graz, Austria, November 20-21, 2008. Proceedings 4*. Springer. 2008, pp. 63–76.
- [110] Tu-Khiem Le et al. “LifeSeeker 2.0: Interactive Lifelog Search Engine at LSC 2020”. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*. Lsc ’20. Dublin, Ireland: Association for Computing Machinery, 2020, pp. 57–62.
-

-
- [111] Nguyen-Khang Le et al. “Smart lifelog retrieval system with habit-based concepts and moment visualization”. In: *Proceedings of the ACM workshop on lifelog search challenge*. 2019, pp. 1–6.
- [112] Hyowon Lee et al. “Constructing a SenseCam visual diary as a media process”. In: *Multimedia Systems* 14 (2008), pp. 341–349.
- [113] Jinhyuk Lee, Seongjun Yun, Hyunjae Kim, Miyoung Ko, and Jaewoo Kang. “Ranking Paragraphs for Improving Answer Recall in Open-Domain Question Answering”. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, 2018, pp. 565–569.
- [114] Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. “Latent Retrieval for Weakly Supervised Open Domain Question Answering”. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, 2019, pp. 6086–6096.
- [115] Jie Lei, Tamara Berg, and Mohit Bansal. “Revealing Single Frame Bias for Video-and-Language Learning”. In: *ACL*. Association for Computational Linguistics, 2023.
- [116] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara Berg. “TVQA: Localized, Compositional Video Question Answering”. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, 2018, pp. 1369–1379.
- [117] Jie Lei, Licheng Yu, Tamara Berg, and Mohit Bansal. “TVQA+: Spatio-Temporal Grounding for Video Question Answering”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, 2020, pp. 8211–8225.
- [118] Andreas Leibetseder and Klaus Schoeffmann. “LifeXplore at the Lifelog Search Challenge 2020”. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*. Lsc ’20. Dublin, Ireland: Association for Computing Machinery, 2020, pp. 37–42.
- [119] Andreas Leibetseder et al. “lifeXplore at the Lifelog Search Challenge 2019”. In: *Lsc ’19*. 2019.
- [120] Mike Lewis et al. “BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, 2020, pp. 7871–7880.
-

-
- [121] Patrick S. H. Lewis et al. “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks”. In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Ed. by Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin. 2020.
- [122] Jiayu Li, Min Zhang, Weizhi Ma, Yiqun Liu, and Shaoping Ma. “A multi-level interactive lifelog search engine with user feedback”. In: *Proceedings of the third annual workshop on lifelog search challenge*. 2020, pp. 29–35.
- [123] Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. “BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation”. In: *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*. Ed. by Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato. Vol. 162. Proceedings of Machine Learning Research. PMLR, 2022, pp. 12888–12900.
- [124] Lin Liao, Dieter Fox, and Henry A. Kautz. “Location-based activity recognition”. In: *Advances in Neural Information Processing Systems 18 [Neural Information Processing Systems, NIPS 2005, December 5-8, 2005, Vancouver, British Columbia, Canada]*. 2005, pp. 787–794.
- [125] Jie Lin et al. “VCI2R at the NTCIR-13 Lifelog-2 lifelog semantic access task”. In: *Proceedings of the 13th NTCIR Conference on Evaluation of Information Access Technologies*. 2017, pp. 28–32.
- [126] Tsung-Yi Lin et al. “Microsoft coco: Common objects in context”. In: *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer. 2014, pp. 740–755.
- [127] Yinhan Liu et al. “Roberta: A robustly optimized bert pretraining approach”. In: *ArXiv preprint abs/1907.11692* (2019).
- [128] Ze Liu et al. “Swin Transformer: Hierarchical Vision Transformer using Shifted Windows”. In: *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*. IEEE, 2021, pp. 9992–10002.
- [129] Jakub Lokoc, Gregor Kovalcik, and Tomas Soucek. “VIRET at Video Browser Showdown 2020”. In: *Lecture Notes in Computer Science*. Springer International Publishing, 2019, pp. 784–789.
- [130] Jakub Lokoc et al. “Is the Reign of Interactive Search Eternal? Findings from the Video Browser Showdown 2020”. In: *ACM Trans. Multimedia Comput. Commun. Appl.* 17.3 (2021).
-

-
- [131] Jakub Lokoč, František Mejzlík, Tomáš Souček, Patrik Dokoupil, and Ladislav Peška. “Video search with context-aware ranker and relevance feedback”. In: *International Conference on Multimedia Modeling*. Springer. 2022, pp. 505–510.
- [132] Jakub Lokoč, František Mejzlík, Patrik Veselý, and Tomáš Souček. “Enhanced SOMHunter for Known-item Search in Lifelog Data”. In: *Proceedings of the 4th Annual on Lifelog Search Challenge*. Lsc’21. 2021, pp. 71–73.
- [133] Jakub Lokoč, Tomáš Souček, and Gregor Kovalčík. “Using an interactive video retrieval tool for lifelog data”. In: *Proceedings of the 2018 ACM Workshop on The Lifelog Search Challenge*. 2018, pp. 15–19.
- [134] Fuchen Long, Ting Yao, Zhaofan Qiu, Xinmei Tian, Jiebo Luo, and Tao Mei. “Learning to localize actions from moments”. In: *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*. Springer. 2020, pp. 137–154.
- [135] Ilya Loshchilov and Frank Hutter. “Decoupled Weight Decay Regularization”. In: *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6–9, 2019*. OpenReview.net, 2019.
- [136] D. G. Lowe. “Object recognition from local scale-invariant features”. In: *Proceedings of the Seventh IEEE International Conference on Computer Vision*. Vol. 2. 1999, 1150–1157 vol.2.
- [137] David G. Lowe. “Distinctive Image Features from Scale-Invariant Keypoints”. In: *International Journal of Computer Vision* 60.2 (2004), pp. 91–110.
- [138] Yingzhou Lu, Minjie Shen, Huazheng Wang, Xiao Wang, Capucine van Rechem, and Wenqi Wei. “Machine Learning for Synthetic Data Generation: A Review”. In: (2023).
- [139] H. Luo, H. Wei, and L. L. Lai. “Creating Efficient Visual Codebook Ensembles for Object Categorization”. In: *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 41.2 (2011), pp. 238–253.
- [140] Mathias Lux and Savvas A Chatzichristofis. “Lire: lucene image retrieval: an extensible java cbir library”. In: *Proceedings of the 16th ACM international conference on Multimedia*. 2008, pp. 1085–1088.
- [141] Chao Ma et al. “Visual Question Answering With Memory-Augmented Networks”. In: *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18–22, 2018*. IEEE Computer Society, 2018, pp. 6975–6984.
-

-
- [142] Anh-Vu Mai-Nguyen, Trong-Dat Phan, Anh-Khoa Vo, Van-Luon Tran, Minh-Son Dao, and Koji Zettsu. “BIDAL-HCMUSLSC2020: An Interactive Multimodal Lifelog Retrieval with Query-to-Sample Attention-Based Search Engine”. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*. Lsc '20. Dublin, Ireland: Association for Computing Machinery, 2020, pp. 43–49.
- [143] Mateusz Malinowski, Marcus Rohrbach, and Mario Fritz. “Ask Your Neurons: A Neural-Based Approach to Answering Questions about Images”. In: *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. IEEE Computer Society, 2015, pp. 1–9.
- [144] Steve Mann. “Wearable computing: A first step toward personal imaging”. In: *Computer* 30.2 (1997), pp. 25–32.
- [145] František Mejzlík, Patrik Veselý, Miroslav Kratochvíl, Tomáš Souček, and Jakub Lokoč. “Somhunter for lifelog search”. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*. 2020, pp. 73–75.
- [146] Tomáš Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. “Distributed Representations of Words and Phrases and their Compositionality”. In: *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*. Ed. by Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger. 2013, pp. 3111–3119.
- [147] Bernd Munzer, Andreas Leibetseder, Sabrina Kletz, Manfred Jurgen Primus, and Klaus Schöffmann. “lifeXplore at the Lifelog Search Challenge 2018”. In: *Lsc '18*. 2018.
- [148] Theodor Holm Nelson. “Complex information processing: a file structure for the complex, the changing and the indeterminate”. In: *Proceedings of the 1965 20th national conference*. 1965, pp. 84–100.
- [149] Son P Nguyen, Dien H Le, Uyen H Pham, Martin Crane, Graham Healy, and Cathal Gurrin. “Vielens, an interactive search engine for lsc2019”. In: *Proceedings of the ACM Workshop on Lifelog Search Challenge*. 2019, pp. 33–35.
- [150] Thao-Nhu Nguyen et al. “LifeSeeker 3.0: An Interactive Lifelog Search Engine for LSC'21”. In: *Proceedings of the 4th Annual on Lifelog Search Challenge*. 2021, pp. 41–46.
- [151] Thao-Nhu Nguyen et al. “LifeSeeker 4.0: An Interactive Lifelog Search Engine for LSC'22”. In: *Proceedings of the 5th Annual on Lifelog Search Challenge*. 2022, pp. 14–19.
-

-
- [152] Thao-Nhu Nguyen et al. “E-LifeSeeker: An Interactive Lifelog Search Engine for LSC’23”. In: *Proceedings of the 6th Annual ACM Lifelog Search Challenge*. 2023, pp. 13–17.
- [153] Thi-Hoa-Cuc Nguyen, Jean-Christophe Nebel, and Francisco Florez-Revuelta. “Recognition of activities of daily living with egocentric vision: A review”. In: *Sensors* 16.1 (2016), p. 72.
- [154] Tri Nguyen et al. “MS MARCO: A Human Generated MACHine Reading COmprehension Dataset”. In: *CoCo@NIPS*. Vol. 1773. CEUR Workshop Proceedings. CEUR-WS.org, 2016.
- [155] Van-Tu Ninh, Tu-Khiem Le, Liting Zhou, Luca Piras, and Michael Riegler. “Overview of ImageCLEFlifelog 2020: Lifelog Moment Retrieval and Sport Performance Lifelog”. In: *CLEF (Working Notes)*. 2020, p. 17.
- [156] Van-Tu Ninh et al. “A baseline interactive retrieval engine for the nticr-14 lifelog-3 semantic access task”. In: *The Fourteenth NTCIR Conference (NTCIR-14)*. 2019.
- [157] Kyosuke Nishida, Itsumi Saito, Atsushi Otsuka, Hisako Asano, and Junji Tomita. “Retrieve-and-Read: Multi-task Learning of Information Retrieval and Reading Comprehension”. In: *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*. Ed. by Alfredo Cuzzocrea et al. ACM, 2018, pp. 647–656.
- [158] Daniel Oliveira and David Martins de Matos. “Transfer-learning for video classification: Video Swin Transformer on multiple domains”. In: *ArXiv preprint abs/2210.09969* (2022).
- [159] Gabriel de Oliveira Barra, Alejandro Cartas Ayala, Marc Bolaños, Mariella Dimicoli, Xavier Giró Nieto, and Petia Radeva. “Lemore: A lifelog engine for moments retrieval at the ntcir-lifelog lsat task”. In: *Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies*. 2016.
- [160] Peter Oram. “WordNet: An electronic lexical database. Christiane Fellbaum (Ed.). Cambridge, MA: MIT Press, 1998. Pp. 423.” In: *Applied Psycholinguistics* 22.1 (2001), pp. 131–134.
- [161] Vicente Ordonez, Girish Kulkarni, and Tamara L. Berg. “Im2Text: Describing Images Using 1 Million Captioned Photographs”. In: *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain*. Ed. by John Shawe-Taylor, Richard S. Zemel, Peter L. Bartlett, Fernando C. N. Pereira, and Kilian Q. Weinberger. 2011, pp. 1143–1151.
- [162] Long Ouyang et al. “Training language models to follow instructions with human feedback”. In: *NeurIPS*. 2022.
-

-
- [163] Jeffrey Pennington, Richard Socher, and Christopher Manning. “GloVe: Global Vectors for Word Representation”. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, 2014, pp. 1532–1543.
- [164] Jesús Andrés Portillo-Quintero, José Carlos Ortiz-Bayliss, and Hugo Terashima-Marín. “A straightforward framework for video retrieval using clip”. In: *Pattern Recognition: 13th Mexican Conference, MCPR 2021, Mexico City, Mexico, June 23–26, 2021, Proceedings*. Springer. 2021, pp. 3–12.
- [165] Eric Prudhommeaux. “SPARQL query language for RDF”. In: <http://www.w3.org/TR/rdf-sparql-query/> (2008).
- [166] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. “Language models are unsupervised multitask learners”. In: *OpenAI blog 1.8* (2019), p. 9.
- [167] Alec Radford et al. “Learning Transferable Visual Models From Natural Language Supervision”. In: *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, 2021, pp. 8748–8763.
- [168] Colin Raffel et al. “Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer”. In: *J. Mach. Learn. Res.* 21 (2020), 140:1–140:67.
- [169] Pranav Rajpurkar, Robin Jia, and Percy Liang. “Know What You Don’t Know: Unanswerable Questions for SQuAD”. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Melbourne, Australia: Association for Computational Linguistics, 2018, pp. 784–789.
- [170] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. “SQuAD: 100,000+ Questions for Machine Comprehension of Text”. In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas: Association for Computational Linguistics, 2016, pp. 2383–2392.
- [171] Siva Reddy, Danqi Chen, and Christopher D. Manning. “CoQA: A Conversational Question Answering Challenge”. In: *Transactions of the Association for Computational Linguistics* 7 (2019), pp. 249–266.
- [172] Mengye Ren, Ryan Kiros, and Richard Zemel. “Image question answering: A visual semantic embedding model and a new dataset”. In: *Proc. Advances in Neural Inf. Process. Syst* 1.2 (2015), p. 5.
-

-
- [173] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks”. In: *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*. Ed. by Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett. 2015, pp. 91–99.
- [174] Ricardo Ribiero, Alina Trifan, and Antonio JR Neves. “MEMORIA: A Memory Enhancement and MOment Retrieval Application for LSC 2022”. In: *Proceedings of the 5th Annual on Lifelog Search Challenge*. 2022, pp. 8–13.
- [175] Stephen Robertson and Hugo Zaragoza. *The probabilistic relevance framework: BM25 and beyond*. Now Publishers Inc, 2009.
- [176] L. Rossetto, R. Gasser, S. Heller, Mahnaz Parian, and H. Schuldt. “Retrieval of Structured and Unstructured Data with vitrivr”. In: *Lsc '19*. 2019.
- [177] Luca Rossetto, Matthias Baumgartner, Narges Ashena, Florian Ruosch, Romana Pernischova, and Abraham Bernstein. “LifeGraph: A Knowledge Graph for Lifelogs”. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*. Lsc '20. Dublin, Ireland: Association for Computing Machinery, 2020, pp. 13–17.
- [178] Luca Rossetto, Matthias Baumgartner, Ralph Gasser, Lucien Heitz, Ruijie Wang, and Abraham Bernstein. “Exploring Graph-querying approaches in LifeGraph”. In: *Proceedings of the 4th Annual on Lifelog Search Challenge*. 2021, pp. 7–10.
- [179] Luca Rossetto, Ralph Gasser, Silvan Heller, Mahnaz Amiri Parian, and Heiko Schuldt. “Retrieval of structured and unstructured data with vitrivr”. In: *Proceedings of the ACM Workshop on Lifelog Search Challenge*. 2019, pp. 27–31.
- [180] Luca Rossetto, Ralph Gasser, Loris Sauter, Abraham Bernstein, and Heiko Schuldt. “A System for Interactive Multimedia Retrieval Evaluations”. In: *MultiMedia Modeling - 27th International Conference, MMM 2021, Prague, Czech Republic, June 22-24, 2021, Proceedings, Part II*. Ed. by Jakub Lokoc et al. Vol. 12573. Lecture Notes in Computer Science. Springer, 2021, pp. 385–390.
- [181] Luca Rossetto, Ivan Giangreco, Ralph Gasser, and Heiko Schuldt. “Competitive Video Retrieval with vitrivr”. In: *Lecture Notes in Computer Science*. Springer International Publishing, 2018, pp. 403–406.
- [182] Bahjat Safadi, Philippe Mulhem, Georges Quénot, and Jean-Pierre Chevallet. “LIG-MRIM at NTCIR-12 Lifelog Semantic Access Task”. In: *12th NTCIR Conference on Evaluation of Information Access Technologies*. Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies. Tokyo, Japan, June 2016.
-

-
- [183] Kuniaki Saito, Andrew Shin, Yoshitaka Ushiku, and Tatsuya Harada. “Dualnet: Domain-invariant network for visual question answering”. In: *2017 IEEE International Conference on Multimedia and Expo (ICME)*. Ieee. 2017, pp. 829–834.
- [184] Jürgen Sauer, Katrin Seibel, and Bruno Rüttinger. “The influence of user expertise and prototype fidelity in usability tests”. In: *Applied Ergonomics* 41.1 (2010), pp. 130–140.
- [185] Roger C Schank and Robert P Abelson. *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*. Psychology Press, 2013.
- [186] Klaus Schoeffmann. “Video Browser Showdown 2012-2019: A Review”. In: *2019 International Conference on Content-Based Multimedia Indexing (CBMI)*. Ieee, 2019.
- [187] Klaus Schoeffmann. “lifeXplore at the Lifelog Search Challenge 2023”. In: *Proceedings of the 6th Annual ACM Lifelog Search Challenge*. 2023, pp. 53–58.
- [188] Abigail J Sellen, Andrew Fogg, Mike Aitken, Steve Hodges, Carsten Rother, and Ken Wood. “Do life-logging technologies support memory for the past? An experimental study using SenseCam”. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. 2007, pp. 81–90.
- [189] Abigail J. Sellen and Steve Whittaker. “Beyond total capture: a constructive critique of lifelogging”. In: *Communications of the ACM* 53.5 (May 2010), pp. 70–77.
- [190] Min Joon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. “Bidirectional Attention Flow for Machine Comprehension”. In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.
- [191] John Paul D. Serrano, Jamie Mitchell A. Soltez, Rodney Karlo C. Pascual, John Christopher D. Castillo, Jumelyn L. Torres, and Febus Reidj G. Cruz. “Portable Stress Level Detector based on Galvanic Skin Response, Heart Rate, and Body Temperature”. In: *2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*. IEEE, 2018.
- [192] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. “Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning”. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, 2018, pp. 2556–2565.
- [193] Jihye Shin, Alexandra Waldau, Aaron Duane, and Björn Þór Jónsson. “PhotoCube at the lifelog search challenge 2021”. In: *Proceedings of the 4th Annual on Lifelog Search Challenge*. 2021, pp. 59–63.
-

-
- [194] Chenglei Si, Shuohang Wang, Min-Yen Kan, and Jing Jiang. “What does BERT Learn from Multiple-Choice Reading Comprehension Datasets?” In: (Oct. 28, 2019).
- [195] Louise N Signal et al. “Children’s everyday exposure to food marketing: an objective analysis using wearable cameras”. In: *International Journal of Behavioral Nutrition and Physical Activity* 14 (2017), pp. 1–11.
- [196] Karen Simonyan and Andrew Zisserman. “Very Deep Convolutional Networks for Large-Scale Image Recognition”. In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015.
- [197] Amanpreet Singh et al. “Towards VQA Models That Can Read”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 8317–8326.
- [198] Robyn Speer, Joshua Chin, and Catherine Havasi. “ConceptNet 5.5: An Open Multilingual Graph of General Knowledge”. In: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*. Ed. by Satinder P. Singh and Shaul Markovitch. AAAI Press, 2017, pp. 4444–4451.
- [199] Florian Spiess and Heiko Schuldt. “Multimodal Interactive Lifelog Retrieval with vitrivr-VR”. In: *Proceedings of the 5th Annual on Lifelog Search Challenge*. 2022, pp. 38–42.
- [200] Christian Szegedy et al. “Going deeper with convolutions”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. IEEE Computer Society, 2015, pp. 1–9.
- [201] Gemini Team et al. “Gemini: A Family of Highly Capable Multimodal Models”. In: (2023).
- [202] Hugo Touvron et al. “LLaMA: Open and Efficient Foundation Language Models”. In: *ArXiv preprint abs/2302.13971* (2023).
- [203] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. “Learning Spatiotemporal Features with 3D Convolutional Networks”. In: *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. IEEE Computer Society, 2015, pp. 4489–4497.
- [204] Ly-Duyen Tran, Thanh Cong Ho, Lan Anh Pham, Binh Nguyen, Cathal Gurrin, and Liting Zhou. “LLQA-Lifelog Question Answering Dataset”. In: *MultiMedia Modeling: 28th International Conference, MMM 2022, Phu Quoc, Vietnam, June 6–10, 2022, Proceedings, Part I*. Springer. 2022, pp. 217–228.
-

-
- [205] Ly-Duyen Tran, Manh-Duy Nguyen, Nguyen Thanh Binh, Hyowon Lee, and Cathal Gurrin. “Myscéal: An Experimental Interactive Lifelog Retrieval System for LSC’20”. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*. LSC ’20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 23–28.
- [206] Ly-Duyen Tran, Manh-Duy Nguyen, Nguyen Thanh Binh, Hyowon Lee, and Cathal Gurrin. “Myscéal 2.0: A Revised Experimental Interactive Lifelog Retrieval System for LSC’21”. In: *Proceedings of the 4th Annual on Lifelog Search Challenge*. 2021, pp. 11–16.
- [207] Ly-Duyen Tran, Manh-Duy Nguyen, Binh Nguyen, Hyowon Lee, Liting Zhou, and Cathal Gurrin. “E-Myscéal: Embedding-Based Interactive Lifelog Retrieval System for LSC’22”. In: *Proceedings of the 5th Annual on Lifelog Search Challenge*. LSC ’22. Newark, NJ, USA: Association for Computing Machinery, 2022, pp. 32–37.
- [208] Ly-Duyen Tran, Manh-Duy Nguyen, Binh T Nguyen, and Cathal Gurrin. “An Experiment in Interactive Retrieval for the Lifelog Moment Retrieval Task at ImageCLEFLifelog2020”. In: *CLEF (Working Notes)*. 2020, p. 12.
- [209] Ly-Duyen Tran, Manh-Duy Nguyen, Binh T Nguyen, and Liting Zhou. “Myscéal: a deeper analysis of an interactive lifelog search engine”. In: *Multimedia Tools and Applications* (2023), pp. 1–18.
- [210] Ly-Duyen Tran, Dongyun Nie, Liting Zhou, Binh Nguyen, and Cathal Gurrin. “VAISL: Visual-aware identification of semantic locations in lifelog”. In: *International Conference on Multimedia Modeling*. Springer. 2023, pp. 659–670.
- [211] Ly-Duyen Tran et al. “An Exploration into the Benefits of the CLIP model for Lifelog Retrieval”. In: *Proceedings of the 19th International Conference on Content-based Multimedia Indexing*. 2022, pp. 15–22.
- [212] Ly-Duyen Tran et al. “Comparing Interactive Retrieval Approaches at the Lifelog Search Challenge 2021”. In: *IEEE Access* 11 (2023), pp. 30982–30995.
- [213] Ly Duyen Tran, Binh Nguyen, Liting Zhou, and Cathal Gurrin. “MyEachtra: Event-Based Interactive Lifelog Retrieval System for LSC’23”. In: *Proceedings of the 6th Annual ACM Lifelog Search Challenge*. 2023, pp. 24–29.
- [214] Minh-Triet Tran, Thanh-Dat Truong, Tung Dinh Duy, Viet-Khoa Vo-Ho, Quoc-An Luong, and Vinh-Tiep Nguyen. “Lifelog Moment Retrieval with Visual Concept Fusion and Text-based Query Expansion.” In: *CLEF (Working Notes)*. 2018.
- [215] Minh-Triet Tran et al. “FIRST - Flexible Interactive Retrieval SysTem for Visual Lifelog Exploration at LSC 2020”. In: *Proceedings of the Third Annual Workshop on Lifelog Search Challenge*. Lsc ’20. Dublin, Ireland: Association for Computing Machinery, 2020, pp. 67–72.
-

-
- [216] Quang-Linh Tran, Ly-Duyen Tran, Binh Nguyen, and Cathal Gurrin. “MemoriEase: An Interactive Lifelog Retrieval System for LSC’23”. In: *Proceedings of the 6th Annual ACM Lifelog Search Challenge*. 2023, pp. 30–35.
- [217] Vijay K. Vaishnavi. *Design Science Research Methods and Patterns: Innovating Information and Communication Technology*. Auerbach Publications, 2007.
- [218] Ashish Vaswani et al. “Attention is All you Need”. In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*. Ed. by Isabelle Guyon et al. 2017, pp. 5998–6008.
- [219] Denny Vrandečić and Markus Krötzsch. “Wikidata: a free collaborative knowledge-base”. In: *Communications of the ACM* 57.10 (2014), pp. 78–85.
- [220] Shuohang Wang et al. “R 3: Reinforced ranker-reader for open-domain question answering”. In: *AAAI*. Vol. 32. 2018.
- [221] Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. “Gated Self-Matching Networks for Reading Comprehension and Question Answering”. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, 2017, pp. 189–198.
- [222] Gemma Wilson, Derek Jones, Patricia Schofield, and Denis J Martin. “The use of a wearable camera to explore daily functioning of older adults living with persistent pain: Methodological reflections and recommendations”. In: *Journal of Rehabilitation and Assistive Technologies Engineering* 5 (2018), p. 2055668318765411.
- [223] Caiming Xiong, Stephen Merity, and Richard Socher. “Dynamic Memory Networks for Visual and Textual Question Answering”. In: *Proceedings of the 33rd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*. Ed. by Maria-Florina Balcan and Kilian Q. Weinberger. Vol. 48. JMLR Workshop and Conference Proceedings. JMLR.org, 2016, pp. 2397–2406.
- [224] Dejing Xu et al. “Video Question Answering via Gradually Refined Attention over Appearance and Motion”. In: *Proceedings of the 2017 ACM on Multimedia Conference, MM 2017, Mountain View, CA, USA, October 23-27, 2017*. 2017, pp. 1645–1653.
- [225] Huijuan Xu and Kate Saenko. “Ask, attend and answer: Exploring question-guided spatial attention for visual question answering”. In: *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*. Springer. 2016, pp. 451–466.
-

-
- [226] Shuhei Yamamoto, Takuya Nishimura, Yasunori Akagi, Yoshiaki Takimoto, Takafumi Inoue, and Hiroyuki Toda. “Pbg at the ntcir-13 lifelog-2 lat, lsat, and lest tasks”. In: *Proceedings of NTCIR-13, Tokyo, Japan* (2017).
- [227] Shen Yan et al. “Video-text modeling with zero-shot transfer from contrastive captioners”. In: *ArXiv preprint abs/2212.04979* (2022).
- [228] An Yang et al. “Enhancing Pre-Trained Language Representations with Rich Knowledge for Machine Reading Comprehension”. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, 2019, pp. 2346–2357.
- [229] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. “Zero-shot video question answering via frozen bidirectional language models”. In: *ArXiv preprint abs/2206.08155* (2022).
- [230] Wei Yang et al. “End-to-End Open-Domain Question Answering with”. In: *Proceedings of the 2019 Conference of the North*. Association for Computational Linguistics, 2019.
- [231] Zhilin Yang et al. “HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering”. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, 2018, pp. 2369–2380.
- [232] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alexander J. Smola. “Stacked Attention Networks for Image Question Answering”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 21–29.
- [233] Yunan Ye, Zhou Zhao, Yimeng Li, Long Chen, Jun Xiao, and Yueting Zhuang. “Video Question Answering via Attribute-Augmented Attention Network Learning”. In: *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017*. Ed. by Noriko Kando, Tetsuya Sakai, Hideo Joho, Hang Li, Arjen P. de Vries, and Ryen W. White. ACM, 2017, pp. 829–832.
- [234] An-Zi Yen, Hen-Hsen Huang, and Hsin-Hsi Chen. “Personal Knowledge Base Construction from Text-based Lifelogs”. In: *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*. Ed. by Benjamin Piwowarski, Max Chevalier, Éric Gaussier, Yoelle Maarek, Jian-Yun Nie, and Falk Scholer. ACM, 2019, pp. 185–194.
-

-
- [235] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. “Coca: Contrastive captioners are image-text foundation models”. In: *ArXiv preprint abs/2205.01917* (2022).
- [236] Zhou Yu et al. “ActivityNet-QA: A Dataset for Understanding Complex Web Videos via Question Answering”. In: *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press, 2019, pp. 9127–9134.
- [237] Changchang Zeng, Shaobo Li, Qin Li, Jie Hu, and Jianjun Hu. “A Survey on Machine Reading Comprehension—Tasks, Evaluation Metrics and Benchmark Datasets”. In: *Applied Sciences* 10.21 (2020), p. 7640.
- [238] Bolei Zhou, Àgata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. “Learning Deep Features for Scene Recognition using Places Database”. In: *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*. Ed. by Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger. 2014, pp. 487–495.
- [239] Liting Zhou, Duc-Tien Dang-Nguyen, and Cathal Gurrin. “A Baseline Search Engine for Personal Life Archives”. In: *Proceedings of the 2nd Workshop on Lifelogging Tools and Applications*. Lta ’17. New York, NY, USA: Association for Computing Machinery, Oct. 23, 2017, pp. 21–24.
- [240] Liting Zhou, Zaher Hinbarji, Duc-Tien Dang-Nguyen, and Cathal Gurrin. “LIFER: An Interactive Lifelog Retrieval System”. en. In: *Proceedings of the 2018 ACM Workshop on The Lifelog Search Challenge*. Yokohama Japan: Acm, 2018, pp. 9–14.
- [241] Pengfei Zhou, Cong Bai, and Jie Xia. “ZJUTCVR Team at ImageCLEFlifelog 2019 Lifelog Moment Retrieval Task”. In: *CLEF (Working Notes)*. 2019, p. 11.
- [242] Qianling Zhou et al. “The use of wearable cameras in assessing children’s dietary intake and behaviours in China”. In: *Appetite* 139 (2019), pp. 1–7.
- [243] Fengbin Zhu, Wenqiang Lei, Chao Wang, Jianming Zheng, Soujanya Poria, and Tat-Seng Chua. “Retrieving and reading: A comprehensive survey on open-domain question answering”. In: *ArXiv preprint abs/2101.00774* (2021).
- [244] Yuke Zhu, Oliver Groth, Michael S. Bernstein, and Li Fei-Fei. “Visual7W: Grounded Question Answering in Images”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 4995–5004.
-