# Performance Evaluation in Multimedia Retrieval

LORIS SAUTER, University of Basel, Switzerland
RALPH GASSER, University of Basel, Switzerland
HEIKO SCHULDT, University of Basel, Switzerland
ABRAHAM BERNSTEIN, University of Zurich, Switzerland
LUCA ROSSETTO, University of Zurich, Switzerland

Performance evaluation in multimedia retrieval, as in the information retrieval domain at large, relies heavily on retrieval experiments, employing a broad range of techniques and metrics. These can involve human-in-the-loop and machine-only settings for the retrieval process itself and the subsequent verification of results. Such experiments can be elaborate and use-case-specific, which can make them difficult to compare or replicate. In this paper, we present a formal model to express all relevant aspects of such retrieval experiments, as well as a flexible open-source evaluation infrastructure that implements the model. These contributions intend to make a step towards lowering the hurdles for conducting retrieval experiments and improving their reproducibility.

CCS Concepts: • **Information systems → Users and interactive retrieval**; **Evaluation of retrieval results**.

Additional Key Words and Phrases: Interactive Multimedia Retrieval, Retrieval Evaluation, Interactive Evaluation, Evaluation System

**ACM Reference Format:**
Loris Sauter, Ralph Gasser, Heiko Schuldt, Abraham Bernstein, and Luca Rossetto. 2024. Performance Evaluation in Multimedia Retrieval. *ACM Trans. Multimedia Comput. Commun. Appl.* 1, 1, Article 1 (January 2024), 23 pages. https://doi.org/10.1145/3678881

## 1 INTRODUCTION

Engaging in experimentation stands as the paramount method for substantiating or refuting hypotheses, thereby pushing the boundaries of human knowledge. Access to precise tools, enabling the formulation, documentation, and reproducibility of experiments, therefore stands as an indispensable element of rigorous scientific inquiry.

While this principle holds true across disciplines, it poses a particularly distinctive challenge in multimedia retrieval and the evaluation of related systems. The particular challenge in this domain manifests in three key aspects: Firstly, multimedia retrieval comes with a diverse set of problems, spanning the straightforward task of locating one or several specific items in a dataset to acquiring aggregations or derivations thereof.

Secondly, multimedia retrieval evaluations can involve both human-in-the-loop (interactive) and automatic, machine-only (non-interactive) settings both for the retrieval process itself [32] and the

Authors' addresses: Loris Sauter, loris.sauter@unibas.ch, University of Basel, Basel, Switzerland; Ralph Gasser, ralph.gasser@unibas.ch, University of Basel, Basel, Switzerland; Heiko Schuldt, heiko.schuldt@unibas.ch, University of Basel, Basel, Switzerland; Abraham Bernstein, bernstein@ifi.uzh.ch, University of Zurich, Zurich, Switzerland; Luca Rossetto, rossetto@ifi.uzh.ch, University of Zurich, Zurich, Switzerland.


1551-6857/2024/1-ART1
https://doi.org/10.1145/3678881

verification of results that follows. The interactivity poses a particular challenge and introduces complexities in both stages.

And finally, the very motivation for research into efficient and effective multimedia retrieval is the sheer growth of multimedia data that could be observed over the past decades, both in volume and variety. The quantity and diversity of which must also be dealt with when conducting multimedia retrieval evaluations.

In this paper, we make three contributions towards a standardization of how multimedia retrieval evaluations are conducted. Firstly, we present a theoretical model formalizing the diverse aspects constituting multimedia evaluation settings in both current and future scenarios. Secondly, we map the concepts outlined in the model to practical applications in real-world evaluation campaigns. And finally, we introduce the open-source Distributed Retrieval Evaluation Server (DRES), which was first demonstrated in [35] and has since been used in multiple interactive multimedia retrieval evaluations [10, 18, 27, 34, 38, 48].

The remainder of this paper is structured as follows: Section 2 provides an overview of the history of performance evaluation in multimedia retrieval and summarizes currently used methods. Section 3 introduces our formal model and Section 4 discusses its applications in real-world scenarios. Section 5 then presents DRES, which we offer as an open-source implementation of the proposed evaluation model. Some practical applications of and use cases for DRES are then discussed in Section 6. Finally, Section 7 offers some outlook and concluding remarks.

## 2 PERFORMANCE EVALUATION IN MULTIMEDIA RETRIEVAL

The field of *information retrieval* and, as such, *multimedia retrieval* is rooted in experimentation and evaluations. That is, evaluations accompanied research in this domain since its inception. The mechanism of evaluating information retrieval methods with a test collection was first introduced by Cleverdon [7] at the College of Aeronautics in Cranfield, UK, back in the 1960s. This series of experiments was initially concerned with appropriate indexing languages for libraries. By creating a reference collection, in which each document was associated with a certain relevance to a pre-defined information need, Cleverdon [7] essentially created what later would become be known as the *Cranfield paradigm* [50]. In these first experiments, verification was a very laborious endeavor since the relevance was based on manual judgments, but it was a breakthrough since the technique allowed for reproducibility. Furthermore, the Cranfield experiments incorporated a concept that these days is referred to *known-item search (KIS)*, meaning that the information need can be satisfied by an item that is known to be contained in the (test) collection. Researchers could run their information retrieval method, e.g., a ranking algorithm, against the same test collection as others, thereby obtaining comparable results. The results were based on metrics that were specifically invented during these experiments, *precision* and *recall* [24], which to this date are commonly used in retrieval evaluations. The two metrics have an inverse relationship, which is why both measures are required for a quality statement [3].

However, precision and recall have been criticized for various reasons, among them that each measure individually has limited expressiveness, that total knowledge of the test collection is required for the calculation, and that neither reflects on the interactivity of modern-day search. The Cranfield paradigm also faced its share of criticism over the years, for instance, that test collections were too small compared to real-world data. Yet, the Cranfield experiments remained the de facto standard for information retrieval evaluations for several decades.

In 1991, efforts were started at the US National Institute of Standards and Technology (NIST) to create a new, standard dataset in the context of the *Text REtrieval Conference (TREC)*. The TREC initiative essentially built on the Cranfield methodology and made adaptions where necessary [16]. Based on a *test collection* of documents, a set of *tasks* (in TREC terminology: *topic*) were formulated.

Participating research groups had to submit their ranked result list for each topic in one batch. Subsequently, a manual assessment of the relevance of each task produced the evaluation results. Other such "batched" evaluation efforts centered around language retrieval were started shortly after TREC. Notably, the Asian counterpart to TREC, the *National Institute of Informatics Test Collection for IR Systems (NTCIR)*, in 1999, has been held every 18 months since then. NTCIR introduced English-Japanese translation tasks [21] and eventually the field of patent retrieval evaluations [20]. Similarly structured to TREC, natural language processing (of Asian languages) has been another central part of NTCIR. In 1998, a document collection from Switzerland that contained articles in three languages, German, French, and Italian, was published [43], which led to the inception of cross-language retrieval tasks (CLIR) in TREC. Finally, CLIR tasks for European languages moved from TREC to its own initiative in 2000: The Cross-Language Evaluation Forum (CLEF, later Conference and Labs of the Evaluation Forum) was born in 2000, including English, French, German, and Italian [11] content.

Several of the aforementioned evaluation campaigns started to include multimedia retrieval tasks over the years. However, the vast majority did not directly consider the interaction between the user and a retrieval system and stuck to the Cranfield paradigm and the batched submission mode. Ultimately, this is a criterion to divide the major multimedia retrieval evaluation efforts today into the categories *non-interactive* and *interactive*.

### 2.1 Non-interactive Multimedia Retrieval Evaluation

In the non-interactive multimedia retrieval domain, the following initiatives have been established:

**TRECVid** [2, 44] is an annual workshop spun off from the Text Retrieval Conference (TREC) in 2003. In the years since, it hosted various tasks related to video retrieval, including Video Instance Search, Copy Detection, Known-Item Search, and Ad-hoc Video Search. For each task, the organizers provide a dataset as well as common metrics for assessing the quality of the results produced and submitted.

**ImageCLEF** [19] is part of the Cross-Language Evaluation Forum (CLEF) and aims to provide *"an evaluation forum for the cross–language annotation and retrieval of images"*.[1] Held annually since 2003, it offers various image retrieval-related tasks from multiple domains, including medical images or environmental photography.

**MediaEval** is an annual multimedia benchmarking initiative[2] hosting a multitude of multimedia analysis tasks, such as image retrieval for news articles [25], media memorability prediction [26], visual sentiment analysis [17], or music emotion recognition [47]. The initiative is quite broad, and not all tasks have clear retrieval aspects.

Apart from these larger campaigns, each with its various tracks and sub-tasks, there exists a multitude of other and more specialized evaluation venues containing retrieval components, which all fall in the non-interactive evaluation category [1, 8]. While these are sufficient to evaluate certain aspects of retrieval methods in isolation, they fail to reflect the end-to-end experience of somebody using a retrieval system in practice.

### 2.2 Interactive Retrieval Evaluation

Central to the Cranfield experiments' user model —as this was a reasonable assumption these days— were indexers or professional (re-)searchers: Human agents familiar with the domain and process of handling large knowledge collections. Other than that, interaction with users who seek precise information through searching —searchers [9, 50]— has been very limited in the early

[1] https://www.imageclef.org
[2] https://multimediaeval.github.io/

days during the Cranfield experiments. Nevertheless, Keen (1978), previously known for his work on metrics in the Cranfield series (cf [24]) started work that focused more on the aspect of the searcher [22, 23]. Major early-day work has been conducted by Belkin [5] and introduced the hypothesis that end users typically are not able to precisely formulate their information need, and as such, user queries could be grouped by requirements to the information retrieval system [4, 5]. These early-day experiments would entirely focus on text retrieval since multimedia retrieval was not yet a dedicated field at the time. TREC incorporated human-in-the-loop processes to some extent ever since its inception, as the ranked result lists to be submitted could also be created manually [50]. In general, these tasks did, however, not explicitly consider interactivity. The first formalized *interactive track* was introduced in TREC 3, aiming at comparing human and automatic routing efforts [33]. Interactive *video* search roots in TRECvid's interactive video search track and opening up towards the field of interactive content-based search [46].

While these evaluation initiatives provide essential and valuable results, they do not systematically consider the search activity itself as part of the evaluation. This paved the ground for the first interactive and *competitive* evaluation efforts designed to —apart from producing the evaluation results— entertain a crowd (the conference participants) and demonstrate interactive (video) retrieval systems in use. A prime and early example of such a campaign was the VideOlympics [45]. Built similarly to TRECvid ad-hoc search tasks, VideOlympics had its infrastructure specifically tailored to the interactivity of the event: an evaluation server displayed a scoreboard of the results compared to a ground truth in real time. The Video Browser Showdown (VBS) [29, 39, 40] to this day follows and expands upon this paradigm by providing tasks in three distinct categories; (i) Textual Known-Item Search (T-KIS), (ii) Visual Known-Item Search (V-KIS), and (iii) Ad-hoc Video Search (AVS) [41]. The first two task types build upon the very same principle of Cranfield's Known-Item Search [7] and only differ in the modality used to present the information need. For T-KIS, a textual representation of the information need is presented, and for V-KIS, the very portion of the video to be found is previewed. AVS tasks are structurally closer to TRECvid in that human judges assess submissions for a short (broad) textual information need. Notably, for AVS, a large and diverse set of results is expected. In its latest installment, in 2024, VBS operated on three distinct data sets: One being very large and diverse in nature – the Vimeo Creative Commons Collection (V3C) [36], and the other two much smaller but highly homogeneous, sporting diving and underwater footage – the Marine Video Kit (MVK) [49] – and videos of endoscopic surgeries.

Competition-style benchmarking campaigns are also used by other multimedia retrieval communities, such as in lifelog retrieval [52] with the Lifelog Search Challenge (LSC) [12, 15]. Similarly to VBS, LSC used T-KIS tasks in addition to Ad-hoc Lifelog tasks in its latest installment. Most recently, Question and Answering tasks were also employed, in which an answer to a question had to be provided based on information found in the LSC dataset [13]. An example of such a question could be *"What's the name of the restaurant I frequently visit when traveling to Oslo?"*. These types of analytics tasks differ in that the solution to the task is not part of the test collection but instead must be derived from its content, in this example, by finding the right items and reasoning about them. Therefore, formally, the solution is a specific derivative of items in the test collection.

In light of these different types of campaigns, some of the more recent work also aims to provide frameworks for interactive (multimedia) retrieval efforts [28] by categorizing the space of tasks that were used in the past and might be used in the future. Additionally, these types of interactive settings also allow for the analysis of aspects that go beyond the mere correctness of the results that have been submitted. For example, [30] leveraged *interaction logging* during the VBS 2018 installment to analyze what features the systems provided were used most frequently and successfully. This study was further built upon in [31], where an in-depth study was performed for the three top-performing systems that participated in VBS 2020.

## 3 A MODEL FOR PERFORMANCE EVALUATION IN MULTIMEDIA RETRIEVAL

Irrespective of the specific domain of information retrieval evaluation, in its most general form, an evaluation can be described as consisting of the following six core components (loosely based on [44] "formula for TREC"):

**Test collection** is a collection of media items —which we will henceforth refer to as *documents*— from which results are produced during an evaluation.

**Tasks** describe an actionable search desire and information need as well as the constraints under which that information need must be fulfilled. Tasks constitute a major building block of every evaluation, which typically consists of a range of tasks that must be solved.

**Agents** are the entities that solve a task. Typically, the agent is a tandem of the human operator and the instance of the retrieval system they interact with. However, other constellations are possible (e.g., machine-only).

**Evaluation run** is a concrete instance of the evaluation with the agents, aiming to solve the tasks. The differentiation between an evaluation and a run implies that —in principle— an evaluation can be run multiple times.

**Relevance judgment** is the process of assigning relevance (either binary or continuous) to the solutions proposed by the agents with regard to each task, e.g., the documents in the test collection.

**Analysis** describes the analysis of the metrics recorded during the evaluation, which is particularly important in comparative studies.

Inspired by the three pillars, *data*, *task* and *user* proposed by [42], we introduce a separation of an evaluation into three phases: The preparation phase before an evaluation is being run, the phase during which the evaluation is executed and the phase after the evaluation has concluded, and results are being analyzed. The individual phases are introduced in the next sections.

### 3.1 Evaluation Definition

Before any evaluation can be executed, it has to be defined in such a way that the execution is actionable. Therefore, we propose a formal model of such a definition in this section. While [42] restricted the first pillar to the test collection an evaluation operates upon, we extend this idea also to include the definition of the tasks. In doing so, we acknowledge the importance of both the test data as well as the careful design of tasks and the associated description of the information need.

In classical, text-based information retrieval, oftentimes, the unit of retrieval is a *document* [16], which is an element of the test collection. However, particularly for *dynamic multimedia data* —that is, multimedia data that exhibits temporal progression such as videos [6]— the unit of retrieval is often based at a sub-document level, for example, a temporal segment of a video. In some instances, the information need may not directly be included in the test collection at all but instead be some derivation of the data (e.g., a piece of information that can be extracted from an image). To accommodate both aspects, we distinguish between the *document* $\gamma \in \Gamma$ as an element of the *test collection* $\Gamma$ and the *fragments* $\omega$ the documents are comprised of. Our model is strictly agnostic regarding the semantics used to form these fragments. Consequently, a fragment can represent any part or derivation of the content of a document, such as a shot in a video, a region in an image, or a piece of information contained in the content. We denote the *fragment set* $\Omega_\Gamma$ as the (potentially infinite) set of all possible fragments $\omega \in \Omega_\Gamma$, given some test collection $\Gamma$.

Central to an evaluation following the Cranfield paradigm is a notion for information need descriptions, also referred to as *search need presentation* [27] and corresponding relevance judgments. As [27] points out, such search need presentations may themselves exhibit a temporal progression and can thus change over time. We, therefore, formalize the *task description* `desc` as a class of

functions that map some notion of time $T$ to a set of fragments. The function's co-domain $\mathcal{P}(\Omega)$ includes the empty set in case no task description exists at a given point in time. Essentially, the desc function presents such search need presentations in the form of *fragments* for a given point in time or none. A concrete example of such a desc function is described and visually represented in Section 4.1.

$$\texttt{desc}: T \rightarrow \mathcal{P}(\Omega) \tag{1}$$

Relevance judgments, sometimes also referred to as the ground truth, indicate which fragments are relevant with respect to an information need description. This is modeled by the *relevance judgment* function rel, which maps an answer set $A$ to a verdict regarding its relevance. The notion of an answer set is properly defined in Equation (6). For now, it can be thought of as a set of fragments an agent considered relevant and thus submitted as an answer for a task. In order to accommodate binary and graded relevance, the semantics of the function's co-domain are as follows: No relevance is indicated by 0, any form of (potentially gradual) relevance by $(0, 1]$. The symbol $\bullet$ denotes *undecidable* relevance, which can be useful in some corner cases.

$$\texttt{rel}: A \rightarrow [0, 1] \cup \{\bullet\} \tag{2}$$

Collectively, the two functions desc and rel form a *task template* $\hat{z}$, which is a tuple as defined by Equation (3). It defines the two fundamental aspects of every task: how the information need is described to the agents and how potential solutions are evaluated against the ground truth.

$$\hat{z} := \langle \texttt{desc}, \texttt{rel} \rangle \tag{3}$$

In addition to these two required elements, a *task template* can also encode arbitrary boundary conditions, such as a pre-defined running duration, as additional fields. A collection of $N$ *task templates* can then be combined into an *evaluation template* $\hat{e}$, which outlines the content of an evaluation.

$$\hat{e} := \{\hat{z}_1, \hat{z}_2, \ldots, \hat{z}_N\} \tag{4}$$

This evaluation template forms the foundation of the next phase – the evaluation's execution.

### 3.2 Evaluation Execution

The evaluation definition phase is followed by its execution (the evaluation run). This phase follows the idea of creating *task instances* $z$ as elements in the set of tasks $Z$ based on the predefined *evaluation template* $\hat{e}$ and the *task templates* $\hat{z}$ it contains, leading to the minimal definition of an *evaluation* $e$in Equation (5).

$$e := \langle \hat{e}, P, Z \rangle \tag{5}$$

Upon creation, the evaluation $e$ contains the evaluation template it has been created from as well as a set of participating agents $P$. The task set $Z$ is empty at the beginning and populated with tasks $t$ as the evaluation progresses. Every task is an instance of a task template and can be regarded as a copy thereof, with extensions required to encode all the information accumulated during an evaluation.[3]

As tasks are being executed, searchable assignments described by the task $z$ are acted upon by the participating agents $p \in P$. Based on the description generated by a task's desc function,

[3]Separating between the definition and the instance has several advantages. Most importantly, it allows for re-using the same information need description in multiple task instances, which may be a practical necessity.

these agents aim to satisfy the information need within the constraints imposed and submit their findings, commonly in the form of a ranked list, which are then evaluated by a task's `rel` function.

We model these result lists as *answer sets* $A$, which are ordered lists of *answer tuples* $a$ that contain a fragment (the proposed solution) and an optional parameter $r$.

$$A := \{a = \langle \omega, r \rangle \, | \omega \in \Omega \wedge \; r \in [0, 1]\} \tag{6}$$

Using Equation (6), the *submission* $s$ provided by an agent $p$ can be defined as the triple outlined in Equation (7), with $t_s$ representing the point in time relative to the start of the task the agent submitted the answer set.

$$s := \langle A, p, t_s \rangle \tag{7}$$

This idea of a submission leads us to the structure of a task $z$ in an evaluation. The task references the task template $\hat{z}$ it has been derived from. To hold all submissions encountered in the course of its execution, the task must also hold a set of all submissions $S$.

$$z := \langle \hat{z}, S \rangle \tag{8}$$

As with the templates, both the task $z$ and the evaluation $e$ may exhibit additional attributes depending on the concrete use case. Equations 5 and 8 therefore constitute the minimal, viable definitions, which may be tailored to a particular application. For example, both evaluations and tasks typically have a defined start and end time in practice, which can be included as fields.

The basic evaluation execution process is illustrated in Figure 1 as a pseudo finite automaton. Generally speaking, once an evaluation is started (after its initial definition), an evaluation goes into a *preparation* state, which may include the distribution of the test collection or other preparatory steps, such as pre-processing of task descriptions. Once prepared, the evaluation becomes active and enters the *running* state, during which tasks are being created from a template, prepared, and executed.

Every time a task is activated, it undergoes a creation and preparation stage. Once preparation of a task has concluded, it enters the *running* state, during which submissions by the agents are accepted. Once a task has finished, it transitions into the *ended* stage, and the next task may begin. Once all tasks have concluded, the evaluation ends. A task's transition from *running* to *ended* is task-specific. For example, a particular task may be considered solved as soon as every participating agent has submitted at least one answer. Alternatively, a task may have a predefined runtime and may end as soon as the time has elapsed. This is, in fact, true for most transitions, which may be either driven by some internal logic of the system or by a human conductor of the evaluation.

With respect to how evaluations are conducted, and based upon the classification of evaluations presented in Section 2, one can (roughly) distinguish between the following three modes:

**Interactive-synchronous Evaluation** At any point in time, every participating agent $p$ is engaged in at most one task $z$ (interactive), which must be identical for all participating agents, including synchronized start and end time (synchronous). Interactive-synchronous evaluations often have a human conductor orchestrating the evaluation. Typically, tasks have a short run time, and collectively, they constitute a progression over the course of an evaluation, which is the same for all agents.

**Interactive-asynchronous Evaluation** At any moment in time, every participating agent is engaged in at most one task $z$ (interactive), which may, however, differ between different agents (asynchronous). For such evaluations, the logic of how the evaluation progresses is typically automated. In practice, the order of tasks might be different for every participating agent. Tasks typically also have a short time span.

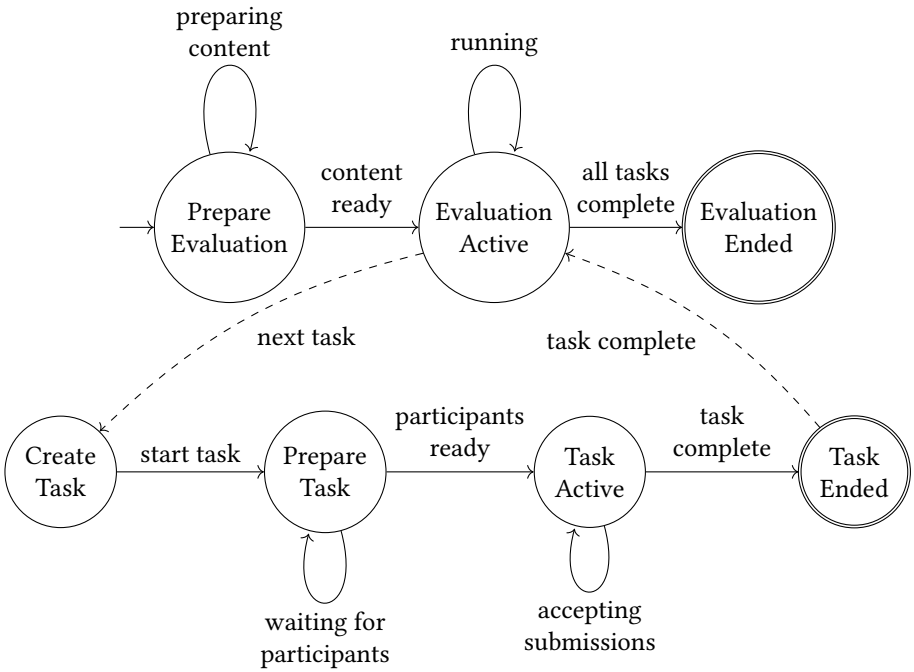



Fig. 1. State-machine representation of the evaluation execution phase. The preceding (preparation) and following (analysis) have been omitted for the sake of brevity.

**Non-interactive Evaluation** At any moment in time, any participating agent can be engaged in multiple tasks (non-interactive). Non-interactive tasks are inherently asynchronous. Typically, such tasks are long-running and can be solved in any particular order. Usually, no human orchestration is involved (aside from the initial definition). These task types resemble the early batched submission evaluations.

### 3.3 Evaluation Analysis

The third and final phase involves the evaluation analysis. It may take place in real-time during the evaluation as well as after the evaluation execution has concluded. In analogy to Seebacher et al.'s [42] third pillar (the user), the evaluation analysis consists of "judging the success or failure of the similarity-based application [the MR system]" [42, p. 327]. In this sense, all the submissions made are analyzed, and conclusions are drawn from the analysis during this phase.

The formal definition of all entities involved in evaluation definition and execution forms a solid foundation for such an analysis. For a multimedia retrieval evaluation, arguably, the *submissions* introduced in Equation (7) are one of the core entities to work with, especially when answering questions about the effectiveness of individual agents. At a high level, we identify two types of metrics that can be derived directly from the entities described. *Task metrics*, as defined in Equation (9), take an agent $p$ and a task $z$ as input and outputs a non-negative, real-valued score that indicates how well the agent $p$ performed in the task (higher is better).

$$f_{task} \colon P \times Z \rightarrow \mathbb{R}_{\geq 0} \tag{9}$$

Aggregations of such task metrics are possible in any form. However, from the perspective of an evaluation, it is highly likely to aggregate over the entire evaluation. Therefore, we propose the class of *evaluation metrics* analog to task metrics, defined per participating agent $p \in P$ and evaluation $e \in E$ in contrast to per task:

$$f_{evaluation} : P \times E \to \mathbb{R}_{\geq 0} \tag{10}$$

Canonically, an evaluation metric $f_{evaluation}$ is expanded to the tasks of an evaluation $e$. For both types of functions, the domain can be extended by an arbitrary number of auxiliary parameters. The types of analyses can, however, be vastly extended by extending the basic data model described thus far. For example, submissions $s$ can also be extended to contain information about user-system interaction. Such interaction logging was shown to be highly valuable when it comes to distinguishing between the performance of a retrieval system as opposed to its human operator.

## 4 PRACTICAL APPLICATIONS OF THE MODEL

The model for performance evaluation described in Section 3 provides a formal framework for multimedia retrieval evaluations. In this section, we revisit parts of that model and approach it from a more practical point of view in order to further illustrate its application in multimedia retrieval evaluation.

### 4.1 Tasks and Information Need Description

With regard to the task description function we have presented in Equation (1), we acknowledge that in a multimedial practice, this formalization is often simplified considerably. The semantics of said function is to describe the relevant information need to a participating agent. We model this mechanism of conveying the relevant information as a set of temporally aligned *channels* with one channel per information type or modality. For every channel, only a single fragment can be presented at a given point in time. While one could imagine arbitrarily many different types of channels conveying different information, in practice, the information is restricted to what can easily be presented using common audio-visual output devices. Commonly used channels include text, image, video, and audio.

To illustrate this, we present an example of a task with a multi-modal information need description in Figure 2. This example task aims to find a video segment in the test collection. The task's description starts with text (first channel), to which, after 30 seconds, an image is added (second channel). At this point in time, agents can leverage both the textual description and the example image to solve the task. After 90 seconds, the text is expanded, and the sound of a door being shut is played in a loop (third channel), which also adds the aural modality. In an interactive setting, such an aural cue could be used by the agent to verify items in the result set. Starting from the three-minute mark, the desired scene could be played as a video (fourth channel), transforming the task from a textual to a visual search task.

It is also to be noted that while the task description may vary from task to task, the overall form of that description is often shared among a set of tasks within an evaluation. For example, in V-KIS tasks, the task description is always a document from the test collection (or parts thereof). To simplify the definition process and implementation aspects, it may, therefore, make sense to introduce an additional abstraction we call *task type*, that captures such common properties shared by multiple templates.

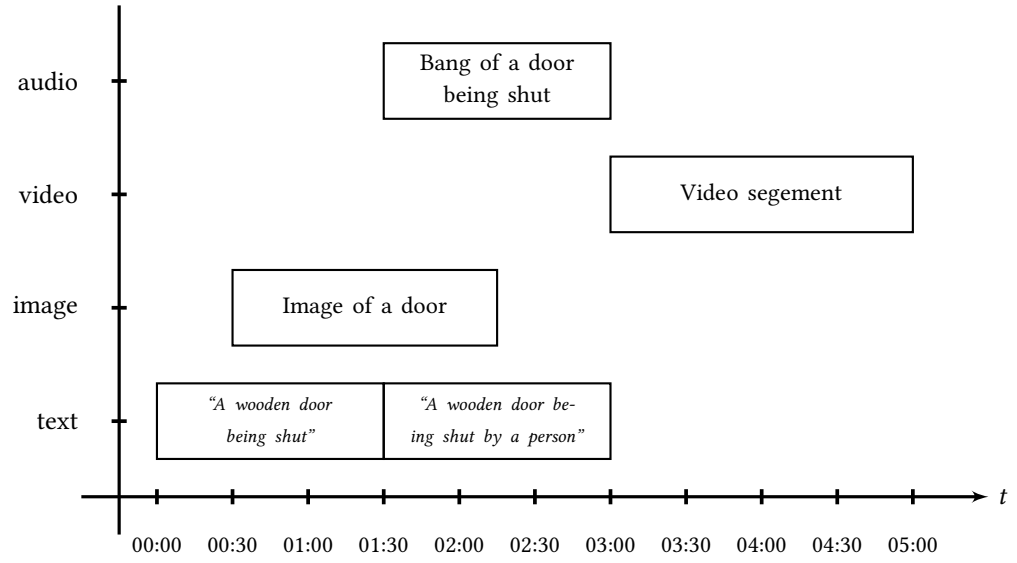



Fig. 2. Illustration of an information need description along different media channels. Rectangles indicate the usage of a channel with a given start and end time. The concept of media channels for task descriptions limits the powerset returned by the task description function described in Equation (1).

### 4.2 Assessment of Relevance

While the relevance function as defined by Equation (2) makes no restrictions on how the relevance of a provided answer is determined, practical instantiations can be categorized along two independent axes, which we will discuss in the following.

Once again, and similar to the task's information need description, the type of relevance assessment is often shared by a set of tasks. Therefore, a similar argument can be made for the introduction of an additional layer of abstraction in the form of task types.

*4.2.1 Time of Determinacy.* The first distinction in the space of relevance functions is whether it is possible for a given task to enumerate possibly relevant answers *a priori* or if it is necessary to render a verdict on a received answer *a posteriori*.

*A priori* relevance assessment requires that for a given query and test collection, a complete annotation of relevance or an otherwise finite set of possible relevant answers exists. This is generally easy for certain types of tasks, such as known-item search scenarios, since only one fragment in the test collection matches the query, assuming the test collection is free of duplicates. For sufficiently small test collections, it might also be feasible to densely annotate the entire collection with respect to the relevance to a particular query beforehand.

*A posteriory* relevance assessment, in contrast, is applicable in all cases where it is not possible or feasible to assess any possible fragment of a test collection or where there might be an arbitrarily large set of relevant answers. Such assessments are generally difficult to mechanize and rather rely on human assessors. An example of a task type that requires this form of assessment is ad-hoc search, where a low-specificity query is given, and it is commonly not feasible nor desirable to densely assess the relevance of the entire test collection.

*4.2.2 Type of Answers.* The second distinction among relevance functions concerns their input. Depending on the task type, the provided answer can consist of *existing* fragments that are part of

the test collection or *derived* fragments that do not necessarily exist before a task is evaluated but are rather created during the evaluation.

*Existing* fragments are commonly used in classical retrieval tasks, where a large amount of content needs to be reduced to a subset relevant to a given information need. The answers for such tasks can be whole documents from a test collection or any part of such documents.

*Derived* fragments, in contrast, are relevant in more analytical tasks, where the answer to a query can not be found in a test collection directly but has to be generated based on relevant information. An example of this answer type is *question answering* tasks, where a question's answer must be given in natural language.

### 4.3 Evaluation Execution

The lifetime of an evaluation run, as depicted in Figure 1, does not further discuss the events triggering any of its state transitions in order to be as generally applicable as possible. In practice, however, there are a few statements that can be made that apply to commonly used scenarios.

A first and rather trivial statement is that the 'next task' transition is generally triggered manually. This is true for both synchronous and asynchronous interactive evaluations, although the actor who triggers the transition is not the same in both cases. For synchronous evaluations, this transition is triggered by the evaluation conductor, while in the asynchronous case, each agent triggers the transition individually. This distinction is also relevant for the following state transitions since the preparation and synchronization step is only explicitly relevant for synchronous cases, where it is important to ensure that all agents can start a task at the same time. In an asynchronous case, each agent's readiness can be implicitly assumed as soon as the 'task start' transition happens. A more in-depth discussion of the differences in state transitions between synchronous and asynchronous evaluations can be found in [37]. For non-interactive evaluations, the 'next task' transition is also triggered by an evaluation conductor, although its semantics are slightly different. Since, in the non-interactive case, all tasks of an evaluation can be active in parallel, this transition happens simultaneously for all tasks rather than one task at a time.

Other broadly applicable statements regarding the 'task complete' transition can be made. For practical reasons, there are several events that can trigger this transition. A commonly used condition is that of a maximum task duration. For interactive evaluations, this duration is commonly defined as the maximum time during which agents can attempt to solve a task (which is part of the task's definition). Non-interactive evaluations might rather have a global submission deadline, defining when answers for any task are accepted. Depending on the type of task and the form of expected answers, there might be additional triggers to end a task before its maximum time is exhausted. This can happen if, for example, all participating agents have already solved the task by providing an answer that is evaluated to be correct. Similarly, a task might only accept a fixed maximum of answers, independently of their correctness, i.e., every agent only gets a single attempt to solve the tasks. For synchronous interactive evaluations, these conditions need to be fulfilled by all agents in order to trigger the premature end of a task. In asynchronous cases, each agent can trigger them individually.

### 4.4 Evaluation Analysis

In order to quantify the performance of any agent, a set of functions is required that represents this performance numerically, as outlined in Section 3.3. Commonly used quantification functions include precision, recall, reciprocal rank, discounted cumulative gain, etc. These measures are commonly defined such that a higher number represents a 'better' performance. They are hence also referred to as *scoring functions*.

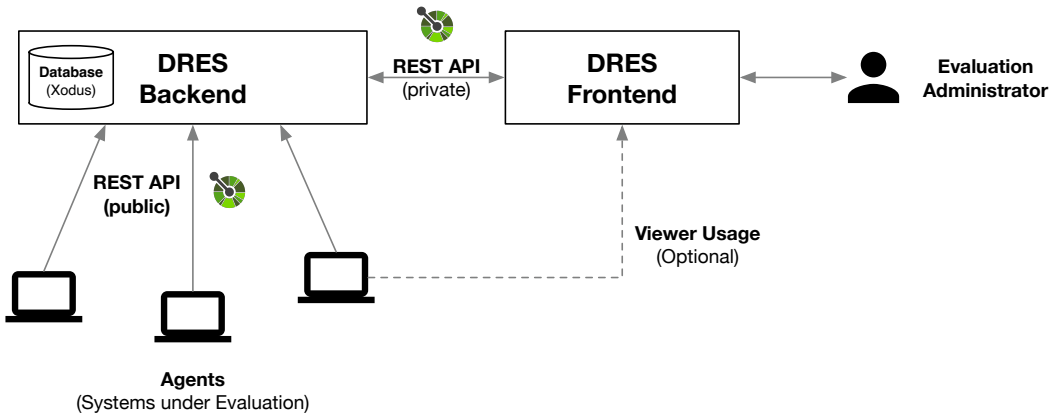



Fig. 3. Overview of DRES' architecture and system components, which include the backend and frontend as well as a persistence layer.

Depending on the task type, more elaborate scoring functions might be used that take into account multiple aspects of provided answers concurrently. For example, for interactive evaluations, it might not only be interesting to evaluate if a provided answer is correct but also how long it took an agent to provide such an answer and/or how many attempts the agent needed before submitting the correct one, penalizing delay as well as mistakes. Scoring functions taking these aspects into account are described in [29], namely for V-KIS, T-KIS, and AVS task types, and used during campaigns such as VBS or LSC.

In addition, and since an evaluation may contain different types of tasks that, in turn, make use of different scoring functions, it may also become necessary to be able to define a set of *aggregation functions*. These are basically scoring functions that build a compound score out of individual scores. Often, these also include normalization aspects, as evaluations commonly compare the relative performance of agents rather than assigning absolute values.

For interactive evaluations, especially synchronous ones, these scoring and aggregation functions are commonly evaluated in real time. This not only has the benefit of providing a continuous readout of the current evaluation state, but it can also be used as a scoreboard showing the current ranking of all agents for information and entertainment. Again, this is common practice in campaigns such as VBS or LSC.

## 5 THE DISTRIBUTED RETRIEVAL EVALUATION SERVER

The theoretical model outlined in Section 3 and the practical aspects described in Section 4, form the foundation for the *Distributed Retrieval Evaluation Server* (DRES) [35]. The implementation of DRES is freely available as open-source software[4] and has already been used for several larger-scale evaluations. Most notably, it has served as the base infrastructure for the annual Lifelog Search Challenge (LSC) [12, 15] since 2020 and for the Video Browser Showdown (VBS) [29, 40] since 2021. This section provides an overview of DRES' architecture and relevant implementation details.

### 5.1 Architecture

DRES is designed as a web application accessible via a browser and it provides all the relevant functionality via a set of REST APIs. This setup was chosen to minimize the requirements on the

[4]https://dres.dev

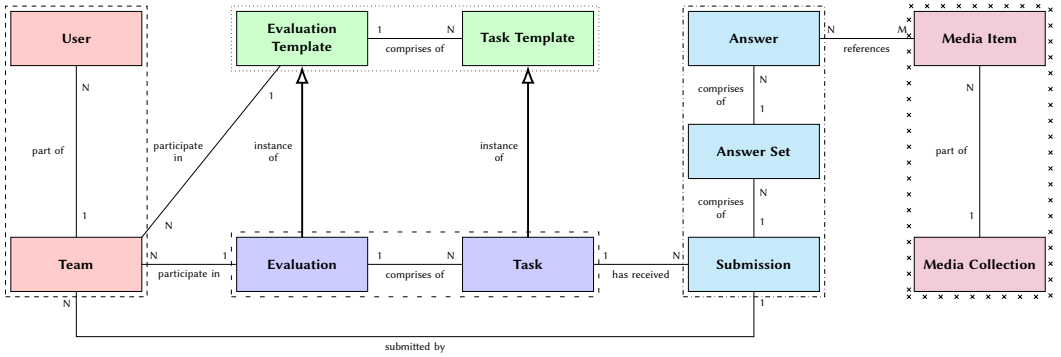



Fig. 4. Overview of DRES' data model. Only the most important entities are illustrated. Entities are grouped as follows: densely dashed group entities related to the agents, entities in the dotted group related to templating (evaluation definition), loosely dashed group entities related to the evaluation execution, dash and dot group entities used to model submissions, and the crossed groups entities representing the test collection.

side of evaluation participants (i.e., the agents) and to facilitate the remote evaluation of retrieval systems in both synchronous and asynchronous settings.

An architecture overview of DRES is provided in Figure 3. DRES consists of two components: a backend application written in Kotlin, which handles the evaluation state and provides API access to all relevant functionality, and a frontend component written in TypeScript and built using the Angular[5] framework, which serves as the primary channel for all user interaction. The state of every evaluation is persisted in a database (JetBrains Xodus[6]).

The browser-based user interface can be used for setup and administration, e.g., by an evaluation organizer. In interactive settings, it is also the main channel for communicating task descriptions to agents and informing participants about the overall progress of an evaluation and its tasks. This also includes features geared towards competitive live settings, such as live score updates.

Agents submit their answers via a public REST API, for which we provide a machine-readable OpenAPI specification. This specification can be used to automate the implementation of client-side stubs, which simplifies the interfacing process with the submission backend, regardless of the client's programming language. The backend- and frontend components also communicate via a (private) REST API, which leverages the same technology.

### 5.2 Data Model

The most important entities in DRES' data model are illustrated in Figure 4. The green entities — namely *evaluation template* and *task template*— are directly related to the evaluation definition phase. These entities can be used to describe and define the evaluations and the tasks they consist of. This includes metadata, the definition of task description and relevance judgement functions and the information about a task's duration. As described in Section 3, these templates are then instantiated into a concrete implementation, which are highlighted in blue — namely *evaluations* and *tasks*. At this level, DRES records information about the start and end of a particular task instance.

The entities marked in red, i.e., *team* and *user*, represent the agents participating in an evaluation: A user is mapped directly to some person or system that can submit to a DRES instance and it is

[5]https://angular.io/
[6]https://github.com/JetBrains/xodus

```
{
  "answerSets": [
    {
      "answers": [
        {
          "mediaItemName": "v-09679",
          "start": 15000,
          "end": 16000
        }
      ]
    }
  ]
}
```

Listing 1. Example of a submission, in which a temporal segment of a single video document that is part of the test collection is being submitted. Specifically, the submission describes video item "v-09679" between second 15 and 16.

also the entity that is used for DRES' authentication and authorization sub-system. Users can be combined into teams if they are meant to cooperate in solving tasks, which affects how submissions are counted and tallied. In that regard, the team is what we consider the agent in the theoretical model, which can consist of one or multiple participating users. This supports the evaluation of collaborative retrieval approaches while still being able to distinguish submissions made by individual team members.

Media collections, i.e., the test collections used for the evaluation, are also mapped to special entities. In fact, DRES offers an entire module dedicated to managing media collections and the items they contain.

### 5.3 Submissions

The entities related to *submissions* are colored in cyan in Figure 4. The data model is very much aligned with the model described in Section 3.2. That is, the submission-related entities directly align to the formal $s$, defined in Equation (7), where the submission entity has a one-to-many relationship with the answer set entity, which in turn has a one-to-many relationship with the answer. Every submission is assigned to the submitting user and team and timestamped upon reception. In deviation from the model and for practical purposes, a single submission can comprise one or many *answer sets*. This allows for optimizations, such as batching multiple answers into a single HTTP request. For this reason, it is also possible to specify the task ID or task name in an answer set in case sets aimed at different tasks are batched together. This is particularly useful for non-interactive evaluation settings. In line with the original definition, an answer set comprises of one to many *answers*, which also allows for more complex scenarios in which not only a single but multiple answers are sought. An individual answer, in turn, can specify either an arbitrary text, a document, or a temporal segment of a document, which covers most but not all of the cases allowed by the definition of a fragment in Section 3.

Individual submissions can be posted to a specified API endpoint, which requires authentication and is, therefore, only available to the users specified in a running evaluation. The data structure of a submission is depicted in Listing 1 as a JSON.

### 5.4 Synchronous and Asynchronous Evaluations

Synchronous evaluations were the first use case for DRES. In the synchronous case, the evaluation administrator serves as a central conductor, determining the start (and end) time of a task, as

well as the task order and the progression within an evaluation. This is an intuitive approach in a localized evaluation setting, where all participants share a common location and, therefore, common experimental conditions. In such cases, task information is commonly presented to all participants using a single instance of the user interface, for example, by projecting it onto a large screen. But even in distributed settings, where each participant requires a separate instance of the user interface, DRES ensures that the state of these instances is synchronized, such that the same information is made available to all participants simultaneously.

In a setting where participants do not share a common location, synchronicity in tasks might be unnecessary or even undesirable, such as when participants are distributed across a larger number of time zones. Such distributed scenarios can benefit participants by enabling them to solve the tasks independently. Therefore, in addition to the *synchronous* interactive evaluation scenario, in which all participants have to solve the same task simultaneously, DRES also supports an *asynchronous* mechanism, where participants can be evaluated independently of each other. This mechanism was first introduced in [37], where it is described in more detail. Asynchronous evaluations are intended for remote settings and grant agents more control over their individual progression within an evaluation, independent of the other agents.

To facilitate this, DRES shows the participant a participant-specific view of the evaluation state, which excludes information about other participants' actions (other than the scores they have achieved). The evaluation administrator is still in charge of defining the tasks of an evaluation but no longer acts as a conductor/coordinator for the evaluation itself. Instead, participants indicate to DRES their readiness for the next task and solve it within the (time-)constraint outlined in the definition. As soon as the task is solved or the time during which solutions are accepted is over, the option to start the next task becomes available. This continues until all tasks of an evaluation have been presented or until the evaluation administrator closes the evaluation. This gives the agent full control over how the evaluation unfolds.

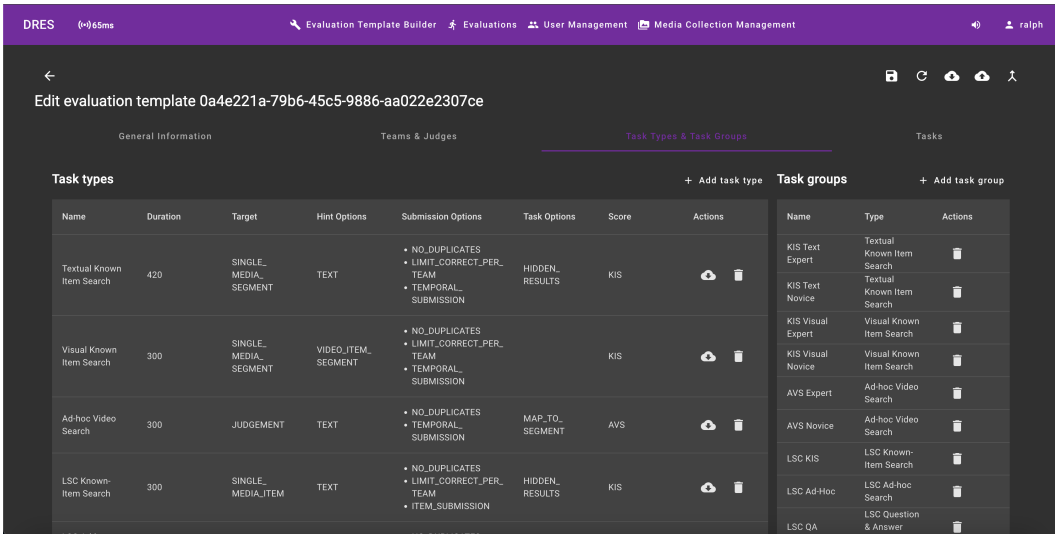
### 5.5 User Interaction

The DRES user interface is mainly geared towards supporting the first two phases of an evaluation life-cycle, namely, the evaluation definition and its execution. However, some basic analysis logic is also included, e.g., in the form of scoreboards that can be presented during a run. The underlying assumption is that the first two phases can be standardized using our proposed model, whereas the analysis is very individual to a particular evaluation setup. Nevertheless, the various data export capabilities and interfaces also offer some support in that regard.

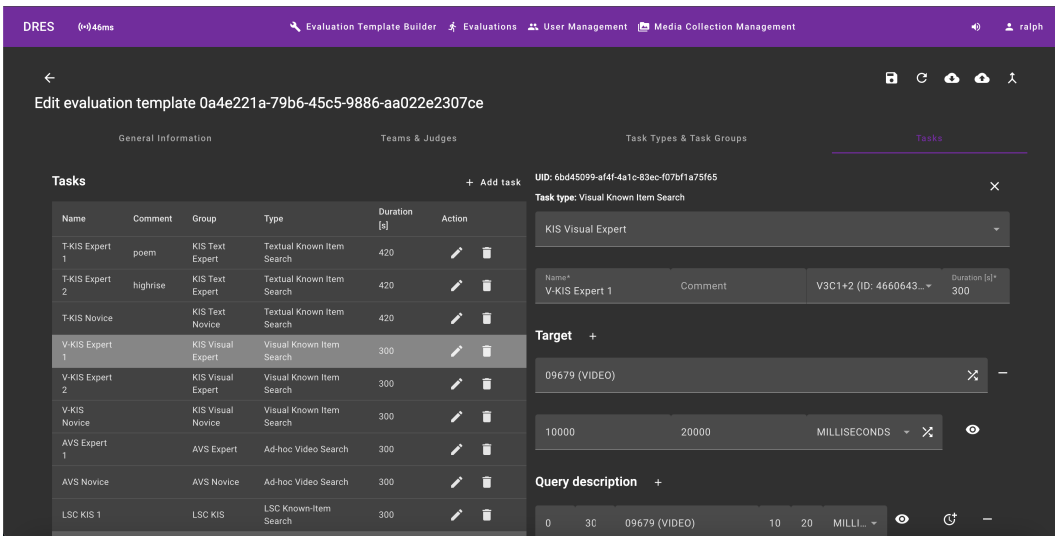
*5.5.1 Evaluation Definition.* DRES' user interface provides a means for evaluation administrators and organizers to manage different evaluation templates and the task templates they contain. The UI for this – the evaluation editor – is depicted in Figure 5. In addition to the meta information that describes the evaluation, the DRES data model supports the definition of the following aspects.

Firstly, an administrator can specify the participating *teams*, which can be made up of different *users* and which can be grouped into *team groups*. The first two entities are described in Section 5.2. Team groups simply provide an additional level of aggregation, mainly geared towards the later analysis stage. In addition to providing this evaluation-specific functionality, users are also the entity for which authentication and authorization are handled. It goes without saying that DRES also offers a user management module for this particular purpose.

Secondly, an administrator can specify *task types* and *task groups*. On the one hand, the task types define attributes shared among multiple tasks as described in Section 4. This is mainly for convenience since every task template inherits said attributes from its type. On the other hand, task




(a) The evaluation editor allows for changes to various aspects of an evaluation template, including the agents participating in the evaluation (teams), the types of tasks, and the individual task templates.




(b) The task editor allows the authoring of individual tasks. The provided examples show a visual KIS task used during the Video Browser Showdown. The relevance judgment is provided by a known ground truth (the target). The task description is a segment of the video in question.

Fig. 5. Screenshots of DRES' evaluation editor, which facilitates the management of evaluation templates.

groups are an organizational feature. One use case during VBS or LSC is to distinguish between the tasks for the main session and those for the novice session of the evaluation.

Lastly, an evaluation organizer can author different task templates using the task editor. This editor's user interface is depicted in Figure 5b. Depending on the type of task description(s) specified

in the task type, the editor can be used to select target items or segments that should be presented to the agents as hints when solving the task. This is complemented by a simple media collection module, in which documents (e.g., images or videos) and temporal segments thereof can be managed. That same module also provides the foundation for specifying the ground truth for relevance judgment. Alternatively, one can also specify textual descriptions or select external resources that can act as hints as well.

*5.5.2 Evaluation Execution.* Once an evaluation has been specified, the DRES user interface can be used to spawn any number of instances for a particular template. Once such an instance has been created, there are two user interface components for the evaluation execution.

On the one hand, the viewer, which is depicted in Figure 6a, is used to communicate the evaluation state to the participating agents. This includes a summary of the scores attained by all the teams. Most importantly, however, the viewer can be used to display the currently active task, the time left for solving it, and the task descriptions that are currently being presented. To ensure fairness in a distributed, synchronous evaluation setting, a synchronization mechanism is implemented so that all the teams are presented with the specified descriptions simultaneously.

The evaluation administration, on the other hand —which is depicted in Figure 6b— gives the evaluation organizer the tools to moderate and orchestrate the evaluation. While useful for all evaluations, it is most useful for synchronous evaluation types. In this setting, the administrator can use the view to switch between tasks, start and end them (see Figure 1), or to increase or decrease the duration of running tasks, should the situation require it. Furthermore, that view can be used to adjust and overrule relevance judgments in case a mistake becomes apparent during the judgment process. All changes made to an evaluation are audited to ensure traceability and reproducibility of results.

## 6 APPLICATION SCENARIOS

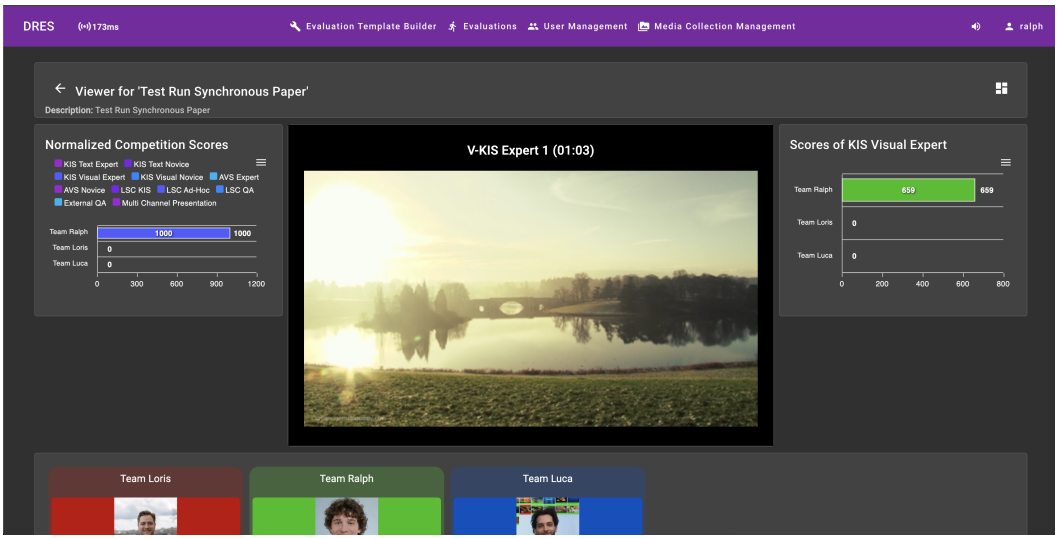
This section starts with providing an overview of evaluation experiments using DRES and continues by presenting several application scenarios demonstrating the practical applicability and usefulness of the introduced model and its implementation in DRES.

### 6.1 Overview of DRES Usage

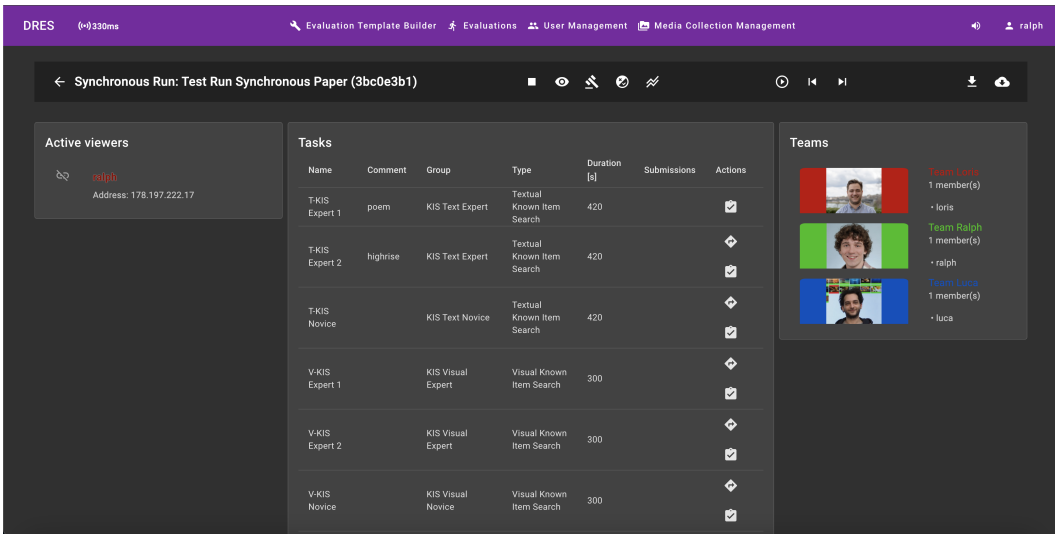
Our implementation of the model, DRES, has been used in both, recurring evaluation benchmarks such as VBS [18, 27] or LSC [12, 15], as well as other challenges, such as the Ho Chi Minh City AI Challenge 2023 [10]. Outside of formal challenge settings, DRES also found several applications in individual experiments [34, 51] and teaching and outreach activities at the Universities of Zurich, Basel, and Charles University in Prague.

Since VBS and LSC are annual evaluation campaigns, we present an overview of the past installments in Table 1. We report on the number of participating teams,[7], the number of individual tasks solved, the number of categories of tasks, as well as the number of individual submissions during the evaluations. For VBS, the reduction of submissions over the last years was by design. Equally so for LSC, the increase in submissions is linked to more categories. For LSC'23 and VBS'24, a new category requiring textual answers not part of the media collection has been introduced.

[7]In VBS'24, a team consisted of one single system operator. However, multiple teams could use the same retrieval system.




(a) The viewer is used to communicate the current state of the evaluation and the active task to all participating agents. This includes information about scores and, most importantly, the active task and its description. In this example, a video is presented (V-KIS). Also, the current scores for all the teams are being tallied and displayed.




(b) The task administration view provides the evaluation organizer the tools to moderate and orchestrate an evaluation. This includes basic functionality such as starting and ending an evaluation but also the ability to control the flow of an evaluation at a task level in synchronous settings.

Fig. 6. Screenshots of DRES' task viewer and task administration view, geared toward evaluation execution.

Table 1. An overview of the past installments of the Video Browser Showdown (VBS) and Lifelog Search Challenge (LSC) using DRES that have been using DRES. For VBS, up until 2024, teams incorporated multiple individual users, whereas, since 2024, there has been a single-user-team policy established.

| | VBS | | | | LSC | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 2021 | 2022 | 2023 | 2024 | 2020 | 2021 | 2022 | 2023 | 2024 |
| Teams | 17 | 16 | 14 | 32 | 14 | 17 | 9 | 14 | 40 |
| Tasks | 37 | 34 | 26 | 34 | 12 | 23 | 26 | 30 | 28 |
| Categories | 3 | 3 | 4 | 15 | 1 | 1 | 3 | 6 | 6 |
| Submissions | 11811 | 18124 | 4452 | 3246 | 192 | 399 | 4373 | 5055 | 7204 |
| Mode | virtual | hybrid | hybrid | hybrid | virtual | virtual | hybrid | hybrid | physical |

### 6.2 Localized Synchronous Evaluations

Comparative experiments of retrieval systems can most easily be performed by having system instances next to each other and using them to solve the same tasks simultaneously in the same environment and under the same conditions. Evaluating systems in such a setting eliminates many possible influences that would otherwise need to be considered. This setting is intuitive and commonly used, from small-scale experiments to large-scale international evaluation campaigns, such as the VBS [29, 40] or the LSC [12, 14, 15]. In 2023 and 2024, DRES served as a base infrastructure, powering both of these campaigns. In each of these challenge instances, different evaluation systems, sometimes with multiple instances per system, were comparatively evaluated on a wide range of tasks, including known-item search tasks with different types of query descriptions, ad-hoc search tasks with live relevance assessment, and question answering tasks.

### 6.3 Distributed Synchronous Evaluations

While it can be desirable to have all systems participating in an evaluation in the same room to ensure a consistent environment for all of them, this is not always easy or even possible. Especially from 2020 to 2022, various pandemic-related travel restrictions made conducting international evaluations at a common location impossible. During that time, the above-mentioned annual campaigns relied on DRES' support for distributed evaluations, such as VBS [18, 27] and LSC [48], to have participants solve the same tasks simultaneously, but with each in a different location. In order to do that, each participant had their own instance of the interface that would present the task information. A simple synchronization mechanism ensures that all participants see the same information simultaneously (within a sufficiently small window of uncertainty), independently of network bandwidth and roundtrip time. A common video call was used as a side-channel for exchanging non-technical information between all participants.

The possibility to run such evaluations in a distributed fashion not only serves as an emergency replacement for on-site experiments it also enables new kinds of evaluations that were not feasible previously. Specifically, it lowers the cost and organizational overhead of scaling up comparative experiments to a larger number of participants. Several such experiments, such as [34, 38], have been conducted so far, independently of the established campaigns.

### 6.4 Distributed Asynchronous Evaluations

When relaxing the locality requirements for comparative experiments, the next possible step is also to relax simultaneity. This is possible due to DRES' support for asynchronous evaluations, where each participant solves tasks independently and not necessarily simultaneously as other

participants. This evaluation scheme enables to have a much larger number of participants, as there is no expensive synchronization overhead. Such a setting has the additional benefit that participants can be recruited independently of each other, such as via a crowdsourcing platform.

We ran initial studies with over 200 participants recruited via prolific[8], which is roughly an order of magnitude increase in participants when compared to the scenarios described above [51]. Such settings, however, easily scale to a much larger number of participants. This opens up possibilities for experiments that were previously not feasible.

### 6.5 Distributed Non-interactive Evaluations

While DRES' primary focus is on evaluation scenarios that require some interactivity, both the model described above and its implementation can also handle the more traditional non-interactive scenarios. Since many existing campaigns rely on bespoke evaluation pipelines for every task, this has the potential of greatly simplifying existing processes and unifying procedures for evaluation participants. Due to its modular architecture, DRES can rather straightforwardly be adapted to many tasks found in current campaigns such as TRECVid, CLEF, or MediaEval.

## 7 CONCLUSION

In this paper, we introduced a formal model for performance evaluation in multimedia retrieval and discussed its practical applicability. We also presented DRES, an open-source evaluation infrastructure capable of supporting a broad range of retrieval evaluations. Due to its flexible and modular architecture, DRES has already been used for a diverse range of retrieval experiments, including several international evaluation campaigns. We also outlined further areas in which DRES might prove useful in the future. With the introduction and the open-source release of DRES, we aim to support retrieval experiments and evaluations in related areas, reduce hurdles for large-scale experiments, and improve reproducibility.

## ACKNOWLEDGMENTS

This work was partially supported by the Swiss National Science Foundation through project MediaGraph (contract no. 202125).

## REFERENCES

[1] Samuel Albanie, Yang Liu, Arsha Nagrani, Antoine Miech, Ernesto Coto, Ivan Laptev, Rahul Sukthankar, Bernard Ghanem, Andrew Zisserman, Valentin Gabeur, Chen Sun, Karteek Alahari, Cordelia Schmid, Shizhe Chen, Yida Zhao, Qin Jin, Kaixu Cui, Hui Liu, Chen Wang, Yudong Jiang, and Xiaoshuai Hao. 2020. The End-of-End-to-End: A Video Understanding Pentathlon Challenge (2020). *CoRR* abs/2008.00744 (2020), 8. arXiv:2008.00744 https://arxiv.org/abs/2008.00744

[2] George Awad, Asad A. Butt, Keith Curtis, Jonathan Fiscus, Afzal Godil, Yooyoung Lee, Andrew Delgado, Jesse Zhang, Eliot Godard, Baptiste Chocot, Lukas Diduch, Jeffrey Liu, Yvette Graham, Gareth J. F. Jones, and Georges Quénot. 2021. Evaluating Multiple Video Understanding and Retrieval Tasks at TRECVID 2021. In *Proceedings of TRECVID 2021*. NIST, USA, NIST, USA, 58. http://www-nlpir.nist.gov/projects/tvpubs/tv21.papers/tv21overview.pdf

[3] Ricardo Baeza-Yates and Berthier A. Ribeiro-Neto. 2011. *Modern Information Retrieval - the concepts and technology behind search, Second edition.* Pearson Education Ltd., Harlow, England. http://www.mir2ed.org/

[4] N.J. BELKIN, R.N. ODDY, and H.M. BROOKS. 1982. ASK FOR INFORMATION RETRIEVAL: PART II. RESULTS OF A DESIGN STUDY. *Journal of Documentation* 38, 3 (mar 1982), 145–164. https://doi.org/10.1108/eb026726

[5] Nicholas J Belkin. 1980. Anomalous states of knowledge as a basis for information retrieval. *Canadian journal of information science* 5, 1 (1980), 133–143.

[6] Henk M. Blanken, Henk Ernst Blok, Ling Feng, and Arjen P. De Vries (Eds.). 2007. *Multimedia Retrieval.* Springer Berlin Heidelberg, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-72895-5

[8] https://prolific.com

[7] Cyril Cleverdon. 1967. The CRANFIELD TESTS ON INDEX LANGUAGE DEVICES. *Aslib Proceedings* 19, 6 (jun 1967), 173–194. https://doi.org/10.1108/eb050097

[8] Keith Curtis, George Awad, Afzal Godil, and Ian Soboroff. 2023. The ACM Multimedia 2023 Deep Video Understanding Grand Challenge. In *Proceedings of the 31st ACM International Conference on Multimedia, MM 2023, Ottawa, ON, Canada, 29 October 2023- 3 November 2023*. ACM, New York, NY, USA, 9606–9609. https://doi.org/10.1145/3581783.3612829

[9] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Ze Wang. 2008. Image retrieval: Ideas, influences, and trends of the new age. *ACM Comput. Surv.* 40, 2 (2008), 5:1–5:60. https://doi.org/10.1145/1348246.1348248

[10] Trong-Le Do, Hai-Dang Nguyen, Quang-Thuc Nguyen, Mai-Khiem Tran, Viet-Tham Huynh, Cathal Gurrin, Tu V. Ninh, Tu-Khiem Le, Thanh Duc Ngo, Tu-Trinh Ngo, Duc-Tien Dang-Nguyen, and Minh-Triet Tran. 2023. News Event Retrieval from Large Video Collection in Ho Chi Minh City AI Challenge 2023. In *Proceedings of the 12th International Symposium on Information and Communication Technology, SOICT 2023, Ho Chi Minh, Vietnam, December 7-8, 2023*. ACM, New York, NY, USA, 1011–1017. https://doi.org/10.1145/3628797.3628940

[11] Nicola Ferro and Carol Peters (Eds.). 2019. *Information Retrieval Evaluation in a Changing World - Lessons Learned from 20 Years of CLEF*. The Information Retrieval Series, Vol. 41. Springer, Cham. https://doi.org/10.1007/978-3-030-22948-1

[12] Cathal Gurrin, Björn Þór Jónsson, Duc Tien Dang Nguyen, Graham Healy, Jakub Lokoc, Liting Zhou, Luca Rossetto, Minh-Triet Tran, Wolfgang Hürst, Werner Bailer, et al. 2023. Introduction to the sixth annual lifelog search challenge, LSC'23. In *Proceedings of the 2023 ACM International Conference on Multimedia Retrieval*. Association for Computing Machinery, New York, NY, USA, 678–679. https://doi.org/10.1145/3591106.3592304

[13] Cathal Gurrin, Klaus Schoeffmann, Hideo Joho, Bernd Münzer, Rami Albatal, Frank Hopfgartner, Liting Zhou, and Duc-Tien Dang-Nguyen. 2019. A Test Collection for Interactive Lifelog Retrieval. In *MultiMedia Modeling - 25th International Conference, MMM 2019, Thessaloniki, Greece, January 8-11, 2019, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 11295)*. Springer, Cham, 312–324. https://doi.org/10.1007/978-3-030-05710-7_26

[14] Cathal Gurrin, Liting Zhou, Graham Healy, Werner Bailer, Duc-Tien Dang-Nguyen, Steve Hodges, Björn Þór Jónsson, Jakub Lokoc, Luca Rossetto, Minh-Triet Tran, and Klaus Schöffmann. 2024. Introduction to the Seventh Annual Lifelog Search Challenge, LSC'24. In *Proceedings of the 2024 International Conference on Multimedia Retrieval, ICMR 2024, Phuket, Thailand, June 10-14, 2024*. ACM, New York, NY, USA, 1334–1335. https://doi.org/10.1145/3652583.3658891

[15] Cathal Gurrin, Liting Zhou, Graham Healy, Björn Þór Jónsson, Duc-Tien Dang-Nguyen, Jakub Lokoć, Minh-Triet Tran, Wolfgang Hürst, Luca Rossetto, and Klaus Schöffmann. 2022. Introduction to the Fifth Annual Lifelog Search Challenge, LSC'22. In *Proceedings of the 2022 International Conference on Multimedia Retrieval*. Association for Computing Machinery, New York, NY, USA, 685–687. https://doi.org/10.1145/3512527.3531439

[16] Donna Harman. 2011. *Information Retrieval Evaluation*. Springer, Cham. https://doi.org/10.2200/S00368ED1V01Y201105ICR019

[17] Syed Zohaib Hassan, Kashif Ahmad, Michael Riegler, Steven Hicks, Nicola Conci, Pål Halvorsen, and Ala I. Al-Fuqaha. 2021. Visual Sentiment Analysis: A Natural Disaster Use-case Task at MediaEval 2021. In *Working Notes Proceedings of the MediaEval 2021 Workshop, Online, 13-15 December 2021 (CEUR Workshop Proceedings, Vol. 3181)*. CEUR-WS.org, Aachen, 3. https://ceur-ws.org/Vol-3181/paper5.pdf

[18] Silvan Heller, Viktor Gsteiger, Werner Bailer, Cathal Gurrin, Björn Þór Jónsson, Jakub Lokoc, Andreas Leibetseder, Frantisek Mejzlík, Ladislav Peska, Luca Rossetto, Konstantin Schall, Klaus Schoeffmann, Heiko Schuldt, Florian Spiess, Ly-Duyen Tran, Lucia Vadicamo, Patrik Veselý, Stefanos Vrochidis, and Jiaxin Wu. 2022. Interactive video retrieval evaluation at a distance: comparing sixteen interactive video search systems in a remote setting at the 10th Video Browser Showdown. *Int. J. Multim. Inf. Retr.* 11, 1 (2022), 1–18. https://doi.org/10.1007/S13735-021-00225-2

[19] Bogdan Ionescu, Henning Müller, Ana-Maria Drăgulinescu, Wen-Wai Yim, Asma Ben Abacha, Neal Snider, Griffin Adams, Meliha Yetisgen, Johannes Rückert, Alba García Seco de Herrera, Christoph M. Friedrich, Louise Bloch, Raphael Brüngel, Ahmad Idrissi-Yaghir, Henning Schäfer, Steven A. Hicks, Michael A. Riegler, Vajira Thambawita, Andrea M. Storås, Pål Halvorsen, Nikolaos Papachrysos, Johanna Schöler, Debesh Jha, Alexandra-Georgiana Andrei, Ioan Coman, Vassili Kovalev, Ahmedkhan Radzhabov, Yuri Prokopchuk, Liviu-Daniel Ştefan, Mihai-Gabriel Constantin, Mihai Dogariu, Jérôme Deshayes, and Adrian Popescu. 2023. *Overview of the ImageCLEF 2023: Multimedia Retrieval in Medical, Social Media and Internet Applications*. Springer Nature Switzerland, Cham, 370–396. https://doi.org/10.1007/978-3-031-42448-9_25

[20] Noriko Kando. 2002. Overview of the Third NTCIR Workshop. In *Proceedings of the Third NTCIR Workshop on Research in Information Retrieval, Automatic Text Summarization and Question Answering, NTCIR-3, Tokyo, Japan, October 8-10, 2002*, Keizo Oyama, Emi Ishida, and Noriko Kando (Eds.). National Institute of Informatics (NII), Tokyo, Japan, 14. http://research.nii.ac.jp/ntcir/workshop/OnlineProceedings3/NTCIR3-OV-KandoN.rev.pdf

[21] Noriko Kando, Kazuko Kuriyama, Toshihiko Nozue, Koji Eguchi, Hiroyuki Kato, and Souichiro Hidaka. 1999. Overview of IR Tasks at the First NTCIR Workshop. In *Proceedings of the First NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition*. National Institute of Informatics (NII), Tokyo, Japan, 11–44. http://research.nii.ac.jp/ntcir/workshop/OnlineProceedings/IR-overview.pdf

[22] E. MICHAEL KEEN. 1973. THE ABERYSTWYTH INDEX LANGUAGES TEST. *Journal of Documentation* 29, 1 (jan 1973), 1–35. https://doi.org/10.1108/eb026547

[23] E. Michael Keen. 1978. *On the Performance of Nine Printed Subject Index Entry Types.* College of Librarianship, Dep. of Information Systems Studies, Aberystwyth.

[24] Micheal Keen. 1966. *Measures and Averaging Methods Used in Performance Testing of Indexing Systems.* Aslib Cranfield Research Project, Cranfield.

[25] Benjamin Kille, Andreas Lommatzsch, Özlem Özgöbek, Mehdi Elahi, and Duc-Tien Dang-Nguyen. 2021. News Images in MediaEval 2021. In *Working Notes Proceedings of the MediaEval 2021 Workshop, Online, 13-15 December 2021 (CEUR Workshop Proceedings, Vol. 3181).* CEUR-WS.org, Aachen, 3. https://ceur-ws.org/Vol-3181/paper2.pdf

[26] Rukiye Savran Kiziltepe, Mihai Gabriel Constantin, Claire-Hélène Demarty, Graham Healy, Camilo Fosco, Alba García Seco de Herrera, Sebastian Halder, Bogdan Ionescu, Ana Matran-Fernandez, Alan F. Smeaton, and Lorin Sweeney. 2021. Overview of The MediaEval 2021 Predicting Media Memorability Task. In *Working Notes Proceedings of the MediaEval 2021 Workshop, Online, 13-15 December 2021 (CEUR Workshop Proceedings, Vol. 3181).* CEUR-WS.org, Aachen, 3. https://ceur-ws.org/Vol-3181/paper10.pdf

[27] Jakub Lokoc, Stelios Andreadis, Werner Bailer, Aaron Duane, Cathal Gurrin, Zhixin Ma, Nicola Messina, Thao-Nhu Nguyen, Ladislav Peska, Luca Rossetto, Loris Sauter, Konstantin Schall, Klaus Schoeffmann, Omar Shahbaz Khan, Florian Spiess, Lucia Vadicamo, and Stefanos Vrochidis. 2023. Interactive video retrieval in the age of effective joint embedding deep models: lessons from the 11th VBS. *Multimedia Systems* 29, 6 (2023), 3481–3504. https://doi.org/10.1007/S00530-023-01143-5

[28] Jakub Lokoc, Werner Bailer, Kai Uwe Barthel, Cathal Gurrin, Silvan Heller, Björn Þór Jónsson, Ladislav Peska, Luca Rossetto, Klaus Schoeffmann, Lucia Vadicamo, Stefanos Vrochidis, and Jiaxin Wu. 2022. A Task Category Space for User-Centric Comparative Multimedia Search Evaluations. In *MultiMedia Modeling - 28th International Conference, MMM 2022, Phu Quoc, Vietnam, June 6-10, 2022, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 13141).* Springer, Cham, 193–204. https://doi.org/10.1007/978-3-030-98358-1_16

[29] Jakub Lokoc, Werner Bailer, Klaus Schoeffmann, Bernd Münzer, and George Awad. 2018. On Influential Trends in Interactive Video Retrieval: Video Browser Showdown 2015-2017. *IEEE Transactions on Multimedia* 20, 12 (2018), 3361–3376. https://doi.org/10.1109/TMM.2018.2830110

[30] Jakub Lokoč, Gregor Kovalčík, Bernd Münzer, Klaus Schöffmann, Werner Bailer, Ralph Gasser, Stefanos Vrochidis, Phuong Anh Nguyen, Sitapa Rujikietgumjorn, and Kai Uwe Barthel. 2019. Interactive Search or Sequential Browsing? A Detailed Analysis of the Video Browser Showdown 2018. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 15, 1, Article 29 (feb 2019), 18 pages. https://doi.org/10.1145/3295663

[31] Jakub Lokoč, Patrik Veselý, František Mejzlík, Gregor Kovalčík, Tomáš Souček, Luca Rossetto, Klaus Schoeffmann, Werner Bailer, Cathal Gurrin, Loris Sauter, Jaeyub Song, Stefanos Vrochidis, Jiaxin Wu, and Björn ÞóR Jónsson. 2021. Is the Reign of Interactive Search Eternal? Findings from the Video Browser Showdown 2020. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 17, 3, Article 91 (jul 2021), 26 pages. https://doi.org/10.1145/3445031

[32] Phuong-Anh Nguyen and Chong-Wah Ngo. 2021. Interactive Search vs. Automatic Search: An Extensive Study on Video Retrieval. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 17, 2, Article 47 (may 2021), 24 pages. https://doi.org/10.1145/3429457

[33] Paul Over. 2001. The TREC interactive track: an annotated bibliography. *Information Processing & Management* 37, 3 (may 2001), 369–381. https://doi.org/10.1016/s0306-4573(00)00053-4

[34] Luca Rossetto, Ralph Gasser, Silvan Heller, Mahnaz Parian-Scherb, Loris Sauter, Florian Spiess, Heiko Schuldt, Ladislav Peska, Tomás Soucek, Miroslav Kratochvíl, Frantisek Mejzlík, Patrik Veselý, and Jakub Lokoc. 2021. On the User-Centric Comparative Remote Evaluation of Interactive Video Search Systems. *IEEE Multimedia* 28, 4 (2021), 18–28. https://doi.org/10.1109/MMUL.2021.3066779

[35] Luca Rossetto, Ralph Gasser, Loris Sauter, Abraham Bernstein, and Heiko Schuldt. 2021. 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 (Lecture Notes in Computer Science, Vol. 12573).* Springer, Cham, 385–390. https://doi.org/10.1007/978-3-030-67835-7_33

[36] Luca Rossetto, Heiko Schuldt, George Awad, and Asad A. Butt. 2019. V3C - A Research Video Collection. In *MultiMedia Modeling - 25th International Conference, MMM 2019, Thessaloniki, Greece, January 8-11, 2019, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 11295).* Springer, Cham, 349–360. https://doi.org/10.1007/978-3-030-05710-7_29

[37] Loris Sauter, Ralph Gasser, Abraham Bernstein, Heiko Schuldt, and Luca Rossetto. 2022. An Asynchronous Scheme for the Distributed Evaluation of Interactive Multimedia Retrieval. In *Proceedings of the 2nd International Workshop on Interactive Multimedia Retrieval* (Lisboa, Portugal) *(IMuR '22).* Association for Computing Machinery, New York, NY, USA, 33–39. https://doi.org/10.1145/3552467.3554797

[38] Konstantin Schall, Werner Bailer, Kai Uwe Barthel, Fabio Carrara, Jakub Lokoc, Ladislav Peska, Klaus Schoeffmann, Lucia Vadicamo, and Claudio Vairo. 2024. Interactive multimodal video search: an extended post-evaluation for the VBS 2022 competition. *Int. J. Multim. Inf. Retr.* 13, 2 (2024), 15. https://doi.org/10.1007/S13735-024-00325-9

[39] Klaus Schoeffmann. 2014. A User-Centric Media Retrieval Competition: The Video Browser Showdown 2012-2014. *IEEE Multimedia* 21, 4 (2014), 8–13. https://doi.org/10.1109/MMUL.2014.56

[40] Klaus Schoeffmann. 2019. Video Browser Showdown 2012-2019: A Review. In *2019 International Conference on Content-Based Multimedia Indexing (CBMI)*. IEEE, IEEE, USA, 1–4. https://doi.org/10.1109/CBMI.2019.8877397

[41] Klaus Schoeffmann, Jakub Lokoc, and Werner Bailer. 2020. 10 years of video browser showdown. In *MMAsia 2020: ACM Multimedia Asia, Virtual Event / Singapore, 7-9 March, 2021*. ACM, New York, NY, USA, 73:1–73:3. https://doi.org/10.1145/3444685.3450215

[42] Daniel Seebacher, Johannes Häußler, Manuel Stein, Halldor Janetzko, Tobias Schreck, and Daniel A. Keim. 2017. Visual Analytics and Similarity Search: Concepts and Challenges for Effective Retrieval Considering Users, Tasks, and Data. In *Similarity Search and Applications - 10th International Conference, SISAP 2017, Munich, Germany, October 4-6, 2017, Proceedings (Lecture Notes in Computer Science, Vol. 10609)*. Springer, Cham, 324–332. https://doi.org/10.1007/978-3-319-68474-1_23

[43] Páraic Sheridan, Jean Paul Ballerini, and Peter Schäuble. 1998. Building a Large Multilingual Test Collection from Comparable News Documents. In *Cross-Language Information Retrieval*. Springer US, Boston, MA, 137–150. https://doi.org/10.1007/978-1-4615-5661-9_11

[44] Alan F. Smeaton, Paul Over, and Wessel Kraaij. 2006. Evaluation Campaigns and TRECVid. In *Proceedings of the 8th ACM SIGMM International Workshop on Multimedia Information Retrieval, MIR 2006, October 26-27, 2006, Santa Barbara, California, USA*. ACM, New York, NY, USA, 321–330. https://doi.org/10.1145/1178677.1178722

[45] Cees G. M. Snoek, Marcel Worring, Ork de Rooij, Koen E. A. van de Sande, Rong Yan, and Alexander G. Hauptmann. 2008. VideOlympics: Real-Time Evaluation of Multimedia Retrieval Systems. *IEEE Multimedia* 15, 1 (2008), 86–91. https://doi.org/10.1109/MMUL.2008.21

[46] Cees G. M. Snoek, Marcel Worring, Dennis C. Koelma, and Arnold W. M. Smeulders. 2007. A Learned Lexicon-Driven Paradigm for Interactive Video Retrieval. *IEEE Transactions on Multimedia* 9, 2 (2007), 280–292. https://doi.org/10.1109/TMM.2006.886275

[47] Philip Tovstogan, Dmitry Bogdanov, and Alastair Porter. 2021. MediaEval 2021: Emotion and Theme Recognition in Music Using Jamendo. In *Working Notes Proceedings of the MediaEval 2021 Workshop, Online, 13-15 December 2021 (CEUR Workshop Proceedings, Vol. 3181)*. CEUR-WS.org, Aachen, 3. https://ceur-ws.org/Vol-3181/paper6.pdf

[48] Ly-Duyen Tran, Manh-Duy Nguyen, Duc-Tien Dang-Nguyen, Silvan Heller, Florian Spiess, Jakub Lokoc, Ladislav Peska, Thao-Nhu Nguyen, Omar Shahbaz Khan, Aaron Duane, Björn Þór Jónsson, Luca Rossetto, An-Zi Yen, Ahmed Alateeq, Naushad Alam, Minh-Triet Tran, Graham Healy, Klaus Schoeffmann, and Cathal Gurrin. 2023. Comparing Interactive Retrieval Approaches at the Lifelog Search Challenge 2021. *IEEE Access* 11 (2023), 30982–30995. https://doi.org/10.1109/ACCESS.2023.3248284

[49] Quang-Trung Truong, Tuan-Anh Vu, Tan-Sang Ha, Jakub Lokoc, Yue Him Wong Tim, Ajay Joneja, and Sai-Kit Yeung. 2023. Marine Video Kit: A New Marine Video Dataset for Content-Based Analysis and Retrieval. In *MultiMedia Modeling - 29th International Conference, MMM 2023, Bergen, Norway, January 9-12, 2023, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 13833)*. Springer, Cham, 539–550. https://doi.org/10.1007/978-3-031-27077-2_42

[50] Ellen M. Voorhees. 2019. The Evolution of Cranfield. In *Information Retrieval Evaluation in a Changing World - Lessons Learned from 20 Years of CLEF*. The Information Retrieval Series, Vol. 41. Springer, Cham, 45–69. https://doi.org/10.1007/978-3-030-22948-1_2

[51] Nina Willis, Abraham Bernstein, and Luca Rossetto. 2024. Task Presentation and Human Perception in Interactive Video Retrieval. *CoRR* abs/2405.04279 (2024), 25. https://doi.org/10.48550/ARXIV.2405.04279 arXiv:2405.04279

[52] Qianli Xu, Ana Garcia Del Molino, Jie Lin, Fen Fang, Vigneshwaran Subbaraju, Liyuan Li, and Joo-Hwee Lim. 2021. Lifelog Image Retrieval Based on Semantic Relevance Mapping. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 17, 3, Article 92 (jul 2021), 18 pages. https://doi.org/10.1145/3446209