Analytic Quality: Evaluation of Performance and Insight in Multimedia Collection Analysis

Zahálka, J.; Rudinac, S.; Worring, M.

DOI
10.1145/2733373.2806279

Publication date
2015

Document Version
Final published version

Published in
MM '15

License
Article 25fa Dutch Copyright Act

Link to publication

Citation for published version (APA):
Zahálka, J., Rudinac, S., & Worring, M. (2015). Analytic Quality: Evaluation of Performance and Insight in Multimedia Collection Analysis. In *MM '15: proceedings of the 2015 ACM Multimedia Conference: October 26-30, 2015, Brisbane, Australia* (pp. 231-240). Association for Computing Machinery. https://doi.org/10.1145/2733373.2806279

Analytic Quality: Evaluation of Performance and Insight in Multimedia Collection Analysis

Jan Zahálka
Intelligent Systems Lab Amsterdam
University of Amsterdam
Amsterdam, The Netherlands
j.zahalka@uva.nl

Stevan Rudinac
Intelligent Systems Lab Amsterdam
University of Amsterdam
Amsterdam, The Netherlands
s.rudinac@uva.nl

Marcel Worring
Intelligent Systems Lab Amsterdam
University of Amsterdam
Amsterdam, The Netherlands
m.worring@uva.nl

ABSTRACT

In this paper, we present *analytic quality* (*AQ*), a novel paradigm for the design and evaluation of multimedia analysis methods. *AQ* complements the existing evaluation methods based on either machine-driven benchmarks or user studies. *AQ* includes the notion of user insight gain and the time needed to acquire it, both critical aspects of large-scale multimedia collections analysis. To incorporate insight, *AQ* introduces a novel user model. In this model, each simulated user, or *artificial actor*, builds its insight over time, at any time operating with multiple categories of relevance. The methods are evaluated in timed sessions. The artificial actors interact with each method and steer the course by indicating relevant items throughout the session. *AQ* measures not only precision and recall, but also throughput, diversity of the results, and the accuracy of estimating the percentage of relevant items in the collection. *AQ* is shown to provide a wide picture of analytic capabilities of the evaluated methods and enumerate how their strengths differ for different purposes. The *AQ* time plots provide design suggestions for improving the evaluated methods. *AQ* is demonstrated to be more insightful than the classic benchmark evaluation paradigm both in terms of method comparison and suggestions for further design.

Categories and Subject Descriptors

H.3.4 [**Information Storage and Retrieval**]: Systems and Software—*Performance evaluation (efficiency and effectiveness)*

General Terms

Measurement; Performance; Experimentation

Keywords

Evaluation; multimedia search and exploration; interactivity; user insight

MM'15, October 26–30, 2015, Brisbane, Australia.
© 2015 ACM. ISBN 978-1-4503-3459-4/15/10 ...$15.00.
DOI: http://dx.doi.org/10.1145/2733373.2806279.

1. INTRODUCTION

The abundance of multimedia collections has given data analysts in diverse fields a rich new resource. For instance, multimedia collections provide forensic evidence in cases of child abuse or terrorism. In many scientific fields, such as physics, new discoveries are made and validated using various observations and simulations, which are decidedly multimedia data. In the business domain, news media companies often find entire stories to cover in the content of social media platforms. In arts, multimedia applications are instrumental in navigating and exploring cultural heritage collections. In most data analytics use cases, timeliness of the results is a critical factor as insight in the data has to lead to an action as fast as possible. For instance, a forensics expert has 48 hours to decide whether a suspect should be detained further. Likewise, a media expert looking for stories connected to a globally-important event like the Charlie Hebdo attack cannot wait weeks before publishing them. In such cases, decisions are based on millions of multimedia data items and thus difficult. In short, analysts from diverse fields of expertise need to gain understanding of increasingly large and complex multimedia collections, and they need to gain this understanding fast. So what is needed to support the analytics process for multimedia collections?

To support multimedia *analytics* tasks, sophisticated underlying multimedia *analysis* tools and techniques are needed. A solid basis of multimedia analysis tool has been presented within the multimedia community, including algorithms like active SVM [33] or tools like Caffe [13]. In recent years, such multimedia analysis methods have reached the state of being able to truly support insightful multimedia analytics: the features are short enough to allow smooth interaction and sufficiently descriptive to allow the algorithms to operate on high semantic levels with high accuracy. In combination with hardware developments, time is ripe for multimedia analytics. However, which multimedia analysis techniques are the best and how to optimize or develop them further for multimedia analytics?

To answer this question we should carefully consider the goal of data analytics: gaining understanding about the data. This understanding, or *insight*, as it is called in the fields of information visualization and visual analytics [31], is complex and requires interplay of a number of factors. Insight builds up on itself and over time, requires all or most data at hand and is often serendipitous [20]. Interaction is thus crucial: the analyst needs to navigate the collection through intelligent interactions in order to gradually build

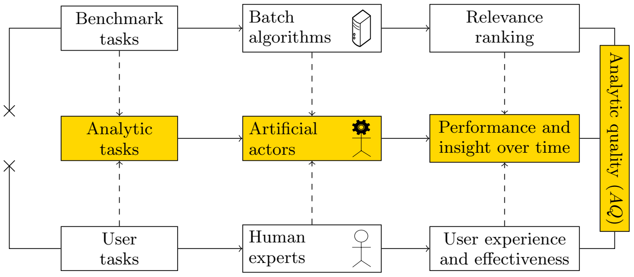

Figure 1: The novelty of analytic quality (*AQ*), the framework proposed in this paper.

insight and not to get overwhelmed by the scale of the data. In the case of multimedia data, analytic tasks involve a combination of exploration and search, and in order to build insight, the analyst needs to be able to organically move back and forth between the two [38]. Hence, it is imperative that techniques behind multimedia analytics support both exploration and search, as well as take into account the various aspects of insight.

Evaluation is a necessary part of designing any good-quality system. The abovementioned works, while providing a theoretical basis for insight, do not provide any means of actually evaluating it. In the multimedia community, we are heavily relying on a number of benchmarks and datasets enabling evaluation of the individual methods. Examples include MediaEval [14], TRECVID [22], MSR-Bing IRC [7], or visual sentiment ontology [1]. These benchmarks are instrumental in establishing, comparing, and improving the quality of the analysis. Most benchmarks focus on the relevance of the results. Indeed, higher relevance improves analytics in general. However, the underlying user model is only implicit and very simple: the notion of relevance is fixed, users process the top ranked results only, and the time for the analysis is unlimited. Do the benchmarks paint the complete picture with respect to the analytic process in which experts are interacting with the data to gain insight?

One way to look at the support of the process of gaining insight is the large body of multimedia research on human-centered interaction and computing [11]. The bulk of the evaluation in this human-centered field is done through user studies, where groups of real users large enough to yield statistically significant results typically assess two well-defined conditions through detailed interviews or questionnaires. This methodology allows for gauging user experience and effectiveness of the evaluated method with regard to the user tasks. From the perspective of multimedia analytics systems design, however, the space of options and possible design paths for future systems is vast and not all these paths can be explored by full user studies. Human-computer interaction studies are appropriate when close to a final design, but different mechanisms are needed in earlier stages.

In order to be truly able to design and assess the analytic potential of multimedia analysis techniques, a new evaluation paradigm is needed. It should not only evaluate the relevance of the returned results, but also the insight gain and the time needed to acquire it. In this paper, we therefore propose *analytic quality (AQ)*, a time-based evaluation framework which evaluates the system performance and at the same time estimates the user insight. *AQ* uses a novel artificial user model, which simulates the behaviour of an analyst striving to gain data understanding. As illustrated in Figure 1, *AQ* brings the worlds of relevance-based benchmarks and user studies closer together. Hence, *AQ* is a significant first step towards evaluation covering both performance and insight aspects of multimedia analytics.

The rest of the paper is organized as follows. Section 2 summarizes the related work. In Section 3, the *AQ* framework is developed along with all its constituent components. Section 4 showcases *AQ* evaluation. Section 5 concludes the paper.

2. RELATED WORK

This section summarizes the related work on evaluation in the multimedia community and the related fields such as computer vision or information retrieval. Namely, we investigate the relevance-based benchmarks, evaluation of interactivity, and existing work on evaluation paradigms going beyond relevance alone.

The dominant paradigm are relevance-based benchmarks. Evaluation sessions are usually not timed and fully automatic with no user involvement. The dominant evaluation metrics are the classic and well-known accuracy, precision, recall, and average precision. These gained traction in the information retrieval benchmarks such as TREC [3] and are now also widely used in computer vision benchmarks, for example ILSRVC [26], Pascal VOC [5], or TRECVID [22]. Some MediaEval tracks also involve these metrics [4]. For a good reason: the main reason for employing a multimedia analysis algorithm is to get results relevant to our intent.

Gaining insight rarely boils down to just going over a list of retrieved results. A visual analytics study by North et al. shows that insight-based evaluation provides richer feedback, and thus much more valuable lessons to the method designers, than the feedback obtained from benchmark-based evaluation [21]. The human cognitive model of visual analytics by Green et al. sheds light on user insight by establishing that a human analyst keeps a categorical model with up to 7 ± 2 categories when reasoning about new data[6]. The carrier of analyst's multidimensional intent is interaction, extensively investigated by Pike et al. [24]. Interactivity is actively researched also by the multimedia community. The key techniques are summarized in a survey by Thomée and Lew [32]. Considering evaluation, interactivity is explicitly considered for example by the Video Browser Showdown, a competition between user-centered video search engines [28]. Extending the visual analytics and interaction theory, Zahálka and Worring model multimedia analytics insight as a set of categories of relevance defined by the analyst herself [38]. However, [38] presents only a theoretical model and does not provide any means of actually evaluating system performance and multimedia analytic quality. All in all, while relevance rightfully takes a spotlight with regard to insight, there are more aspects that contribute to the user's understanding of the data.

The multimedia-related fields are well-aware of the imperfections of relevance alone. Indeed, the gap between relevance metrics and user preferences was confirmed by Sanderson et al. [27]. Numerous metrics expand the classic binary relevance paradigm, mostly originating in the field of information retrieval. The notion of graded relevance is embodied in discounted cumulative gain (DCG) by Järvelin and Kekäläinen [12]. The expected reciprocal rank (ERR) of Chapelle et al. extends graded relevance to include a sim-

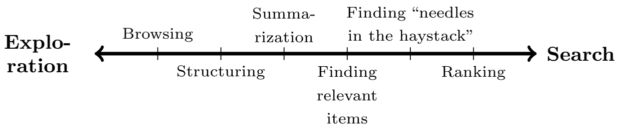

Figure 2: The exploration-search axis with example multimedia analytics (sub)tasks [38].

ple user model, measuring retrieval quality as the inverse expected effort by the user to satisfy her information need [2]. Smucker and Clarke introduce the crucially important notion of time to IR effectiveness measures, making the notion of information gain dependent on time the user needs to reach the item, rather than its position in the list alone [29]. By modelling the correlation between the position of the item in the list and the time the user needs to reach it, this work is one of the first ones to consider time explicitly. These metrics are already being used in text-based benchmarks, including a number of TREC tracks, for example the federated Web search track [3]. Research on evaluation reaching beyond simple benchmark relevance is thus gaining momentum in recent years.

Multimedia analytics tasks consist not only of search, but also of exploration. Hence, to truly evaluate the analytic capabilities of multimedia analysis methods, the exploration component has to be taken into account. In the multimedia domain, exploration is strongly tied to summarization: the analyst gains understanding about the structure of the collection by seeing relevant, representative, and diverse results. The metrics evaluating the quality of summarization again originate in the field of information retrieval, the most prominent examples including BLEU by Papineni et al. [23], ROUGE by Lin [18], Meteor by Lavie and Agarwal [15], and pyramid score by Nenkova et al. [19]. CIDEr by Vedantam et al. automatically measures image description consensus [35]. Some of those metrics have been adapted by the multimedia community. VERT by Li and Merialdo is an extension of BLEU and ROUGE [17], while Rudinac et al. have extended the pyramid score to visual summaries [25]. The increasing importance of diversity is reflected in a number of benchmarks, such as ImageCLEF 2009 [16] or MediaEval Diverse Social Images [9]. The latter resulted in Div400, a diversity benchmark dataset [10].

Overall, the research on evaluation metrics for multimedia is active. There are metrics covering aspects of either search or exploration. Yet multimedia analytics is an intricate interplay of *both*. The main drawback we perceive in most current paradigms is the absence of interaction. Currently, each evaluated method analyzes the entire collection using a relevance indication (e.g., a query or class annotation) and returns a list of results, the entirety of which is used to compute the respective relevance metric. In addition, the notion of relevance is static from the beginning to the end. Time is rarely considered, and few approaches take into account the time the analyst needs to invest to reach a particular result. These shortcomings, combined with the fact that gaining analytical insight is an open-ended task involving both exploration and search in all its phases, is the motivation for our evaluation method developed further in the paper.

3. METHOD

This section presents the *AQ* evaluation framework. Section 3.1 discusses the analytic task model and its implications for *AQ*. Section 3.2 describes the artificial actors, i.e., the user model used by *AQ*. Section 3.3 outlines the details of the *AQ* evaluation process. Section 3.4 presents the evaluation measures used by *AQ*. Finally, Section 3.5 provides guidelines for interpreting the results of *AQ* evaluation.

3.1 Evaluating analytic tasks

Our evaluation method, *AQ*, is built on the notion of the exploration-search axis [38], illustrated in Figure 2. A user analyzing a collection builds insight over time, i.e., her needs, intent, and the notion of relevance are changing over the course of the analysis. Moreover, the user will be tilting back and forth between exploration and search as the insight builds up. To complete her analytic task, the user will be *interacting* with the collection, undertaking (sub)tasks based on the *current* notion of relevance. Examples of these (sub)tasks are depicted on the exploration-search axis in Figure 2. *Analytic categorization*, i.e., the task of assigning individual items into categories defined by the analyst, is the umbrella task for the exploration-search axis task model [38]. These categories can be completely different than any categories associated with the dataset itself, such as class labels or annotations. An example of such a category could be for example "suspicious activity" in forensic research. This category can encode multiple aspects in multiple modalities: e.g., presence of firearms in the image, text description inciting terrorism, or geo location corresponding to terrorist training camps. In order to support the evaluation of analytic tasks, the artifical actors in the *AQ* user model need to be able to create analytic categories of relevance, further denoted simply "categories."

To define categories, an artificial actor needs to "make sense" of the content of the individual items in the analyzed collection. For the individual artificial actors to be able to do that, the items need to be annotated with content annotations, e.g., "this image contains a person, a car, and a house", or "the topics in the text are politics and USA." However, in the analytic context, we cannot expect the collection to be annotated. Hence, we need to collect these annotations ourselves.

To this end, we employ a so called *arbiter*, i.e., a black box producing content annotations for each item in the analyzed collection. An arbiter needs to be:

- *Consistent.* Inputting the same item twice yields identical annotations.
- *Semantic.* A human can interpret the annotations and judge their presence in the item.
- *Autonomous.* The arbiter is self-contained, providing the annotations without any involvement of the evaluated methods.
- *Mostly accurate.* The error rate of the arbiter annotation assignment is as low as possible.

The *consistency* condition prevents randomness and ensures repeatability of the results. The *semantic* condition is necessary for *AQ* to simulate actual human behaviour. It would be technically possible for artificial actors to operate on low-level machine features, but given that humans rarely think in terms of these features, such approach would hardly measure analytic quality. The *autonomous* condition ensures fairness: an artificial actor explicitly using the same data model as one of the evaluated methods would have an innate edge over the other evaluated methods. The requirement for autonomy does not ensure full independence of the annotations. For example, it may occur that a visual arbiter's concept dictionary overlaps with that of one (or more) evaluated methods. Given that the arbiter is autonomous, the underlying features might be correlated, but are not identical. Bearing in mind that this correlation is hard to measure, we conjecture that having non-identical data representations is enough to treat the method's data model as autonomous from the arbiter data model for the purposes of evaluation. The fourth condition requires that the arbiter annotations have to be *accurate* enough. This condition is also hard to assert. However, we conjecture that the level of the state of the art in visual and text analysis algorithms is high enough to provide meaningful annotations. This has been shown by a number of applications, for example in interactive venue recommendation [37].

The arbiter annotation process itself has two steps. The first one is selecting the arbiter. There is an abundance of excellent tools allowing fast and meaningful content annotation. In the visual domain, Caffe by Jia et al. can be used to obtain concepts from a convolutional deep network trained on another dataset [13]. In the text domain, a solid option for content annotation is extracting the LDA topics using the Gensim framework [36]. The second step involves crisp assignment of content annotations to individual items. If the arbiter produces scores for each annotation, these need to be thresholded to determine annotations present on each item. An example of such crisp assignment is assigning those concept annotations reaching at least 80% of the maximum confidence score for each item. Once each item is associated with arbiter annotations, the candidate analytic categories for the actors can be generated.

The candidate categories (C_c) are created from the arbiter annotations as follows. Let $\mathcal{A}_i = \{a_1, a_2, \cdots, a_{|\mathcal{A}_i|}\}$ be the set of visual, text, and metadata annotations associated with item i in the analyzed multimedia collection I. Each category $C \in C_c$ is composed of the arbiter annotations associated with the category ($\mathcal{A}_C$) and the set of all items belonging to the category (denoted I_C):

$$C = \{\mathcal{A}_C, I_C\} \tag{1}$$

The categories of each item $i \in I$ correspond to all subsets of $\mathcal{A}_i$ except the empty set. Indeed, if we take for instance an item i with $\mathcal{A}_i = \{\texttt{blood}, \texttt{firearms}\}$, we want to associate it not only with the category $\{\texttt{blood}, \texttt{firearms}\}$, but also the category $\{\texttt{blood}\}$ and the category $\{\texttt{firearms}\}$. Let $\mathcal{P}(\mathcal{A}_i)$ denote the power set of $\mathcal{A}_i$ and $\mathcal{A}_I = \bigcup_{i \in I} \mathcal{P}(\mathcal{A}_i)$. C_c is then:

$$C_c = \bigcup_{\mathcal{A}_C \in \mathcal{A}_I \setminus \varnothing} \{\mathcal{A}_C, \{i \mid i \in I \wedge \mathcal{A}_C \subseteq \mathcal{A}_i\}\} \tag{2}$$

If necessary, C_c can be further pruned to contain only those categories $C \in C_c$ for which $|I_C|$ is above a certain threshold. All actors will draw their categories from C_c. In the further text, random draw from C_c is a shorthand for uniformly drawing an item from the set and removing it. The removal is done to foster the diversity of the individual artificial actors. The resulting set of candidate categories C_c is the source of analytic categories for the artificial actors described in the next section.

3.2 Artificial actors

The *AQ* user model operates with *artificial actors*, i.e., computer agents interacting with the evaluated method and simulating user behaviour. The artificial actors are built on three cornerstones: analytic categories, changing notion of relevance, and limited time. The analytic categories are obtained from the candidate categories set (C_c) defined in Section 3.1. The actors change their categories of relevance over time, modelling the dynamic nature of insight. The limited time adresses the real-life need for timely analysis. The conjunction of these factors addresses the aspects of insight by North [20]. Providing arbiter annotations of multiple modalities and defining analytic categories as compounds of these annotations simulates *complexity*, since the actors are using all of the data channels or at least most of them. The artificial actor model allows for both incremental evolution of the categories (*depth*) and abrupt category changes (*serendipity*). *Relevance* is embodied in the very mode of interaction, where each actor guides the evaluation session based on what items are relevant to the current category definitions. In this section, we describe the four-step probabilistic process of generating the artificial actors.

Step 1: Initial actor categories. In the first step, the set of categories of relevance (C^a) is established for each actor $a \in A$. As mentioned in Section 2, a study by Green et al. indicates that a human analyst can operate with up to 7 ± 2 categories of relevance [6]. Each artificial actor's number of initial categories is thus an integer uniformly drawn from the $[1, 9]$ interval. Each of the initial categories is in turn uniformly drawn from C_c.

Step 2: Number of insight changes. Once the initial C^a is established, we need to determine the number of insight changes, or *breakpoints*, the actor will make throughout the session (denoted as n_B). The maximum number of these breakpoints ($n_B^{\max}$) is a parameter to be chosen based on the domain of expertise: for example, a casual user browsing celebrity photos will typically have a shorter attention span and focus than a medical scientist analyzing a medical dataset, resulting in a higher $n_B^{\max}$. In our method, we account for different kinds of users within the same domain of expertise. To this end, we propose a simple user behaviour model. A portion of the actors is *single-minded*, with a clear purpose from start to end, i.e., $n_B^{\max} = n_B = 0$. The rest is *volatile* and have their n_B determined in a probabilistic manner. We treat a breakpoint as a rarely occurring event, and hence draw n_B from an exponential distribution with $\lambda = \frac{\ln 10}{n_B^{\max}}$. This distribution ensures that increasingly large values of n_B are drawn with decreasing probability (a number of volatile actors will actually be single-minded with $n_B = 0$). The value of λ ensures that 90% of the actors will have their $n_B < n_B^{\max}$. The remaining 10% accounts for the unpredictability of user behaviour; we cannot assume a crisp upper bound for n_B in any domain.

Step 3: Insight change times. For each breakpoint $b \in \{b_1, \ldots, b_{n_B}\}$, we need to determine the time when it occurs (t_b). The value of t_b is again probabilistic: since insight is serendipitous, occuring unexpectedly [20], it is not fully predictable. However, insight is also deep and it takes time to build it [20]. Thus, given n_B, we can expect the session to be divided in $n_B + 1$ segments of roughly equal length. Let $t^{\text{eq}}_{seg} = \frac{t_s}{n_B+1}$ denote the length of each segment assuming the breakpoints are equidistant. For the i-th breakpoint b_i, we draw t_b from a normal distribution $\mathcal{N}_i(\mu_i, \sigma)$, setting $\mu_i = i \cdot t^{\text{eq}}_{seg}$ and $\sigma = \frac{1}{6} t^{\text{eq}}_{seg}$. The value of μ_i expresses the centering of individual breakpoints on equidistant time ticks. The value of σ is motivated by the desire to leave sufficient time between breakpoints to build up insight (the "deep" characteristic). To assert sufficient segment length, we want the majority of the t_b draws to fall within $\frac{1}{2} t^{\text{eq}}_{seg}$ seconds of the mean. Using the three sigma rule, setting $3\sigma = \frac{1}{2} t^{\text{eq}}_{seg}$ ensures that 99.7% of the drawn values fall within the desired time interval.

Step 4: Insight change actions. The last aspect of the actor's insight scenario to be determined is the type of the insight change and the associated action. These actions have to account for both depth and serendipity of building insight: the changes can range from incremental to abrupt [20]. Thus, in our actor model, we account for 6 distinct insight change events with equal probabilities for each breakpoint:

Action 1: Add category. If $|C^a| < 9$, a category is randomly drawn from C_c and added to the C^a. Otherwise, "replace category" is performed.

Action 2: Remove category. If $|C^a| > 1$, a category is uniformly drawn from C^a and removed from the set. Otherwise, "replace category" is performed, with the removal step being enforced.

Action 3: Replace category. Performs "remove category" followed by "add category."

Action 4: Expand category. Replaces category $C \in C^a$ with a category whose associated annotations are a superset of those associated with C. For example, the {`dog`} category will get replaced by the {`dog`, `house`} category. C^{sup} is established as the set of "annotation superset" categories:

$$C^{\text{sup}} = \{C_k \in C_c \mid \exists C \in C^a : \mathcal{A}_C \subset \mathcal{A}_{C_k}\} \tag{3}$$

Then, a category is uniformly drawn from C^{sup}, replacing that category in C^a whose annotation superset it corresponds to. If there are multiple such categories in C^a, one is selected randomly with uniform probability. If $C^{\text{sup}} = \varnothing$, "add category" is performed instead.

Action 5: Reduce category — Replaces category $C \in C^a$ with a category whose associated annotations are a *sub*set of those associated with C. For example, the {`forest`, `river`} category will get replaced by the {`forest`} category. The reduction process is an analogy to the expand process: C^{sub} is established as the set of "annotation subset" categories:

$$C^{\text{sub}} = \{C_k \in C_c \mid \exists C \in C^a : \mathcal{A}_C \supset \mathcal{A}_{C_k}\} \tag{4}$$

Then, a category is uniformly drawn from C^{sub} to replace a category whose annotation subset it corresponds to. If there are multiple candidates for replacement, the reduced category is again uniformly drawn. If $C^{\text{sub}} = \varnothing$, *remove category* is performed instead.

Action 6: Change category — Replaces category $C \in C^a$ with a category $C_i \in C_c$ whose associated annotations are of the same size and contain at least one annotation associated with category C. For example, the {`parrot`, `rainforest`} category will get replaced by the {`parrot`, `savanna`} category. C^{ch}, the set of candidate replacements, is formally defined as:

$$C^{\text{ch}} = \left\{ C_k \in C_c \mid \exists C \in C^a : \begin{array}{c} |\mathcal{A}_C| = |\mathcal{A}_{C_k}| \\ \mathcal{A}_C \cap \mathcal{A}_{C_k} \neq \varnothing \end{array} \right\} \tag{5}$$

Once C^{ch} is established, an existing category $C \in C^a$ is replaced. If $C^{\text{ch}} = \varnothing$, *replace category* is performed instead.

The artificial actors capture all key aspects of the multimedia analytics process. They operate with multiple categories of relevance, which are not defined by the evaluated method, but externally, using arbiter annotations. This simulates the real user's ability to define the categories herself. Moreover, the artificial actors address all enumerable characteristics of insight as defined by North [20]. The artificial actor model thus provides a user model conforming to multimedia analytics task and interaction theory.

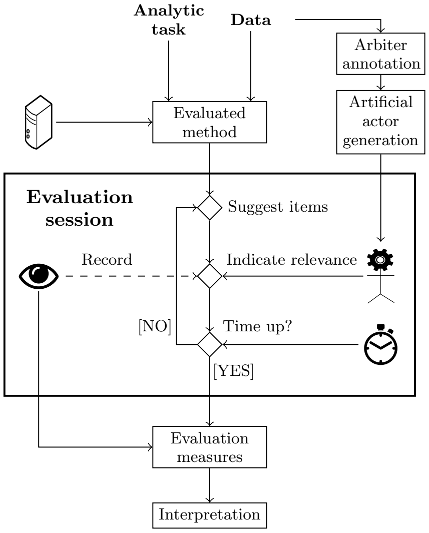

Figure 3: The *AQ* evaluation method pipeline.

3.3 Evaluation pipeline

The pipeline of *AQ* is depicted in Figure 3. In the preparation phase, we need to generate candidate categories of relevance (cf. Section 3.1) and construct the artificial actors (cf. Section 3.2). Each method is evaluated using the same session time (t_s) and the same set of actors (A, the number of actors to be generated is denoted n_A). The value of t_s depends on the domain and/or purpose of the evaluated method. For instance, the sessions in a system for casual exploration of social network content will typically be shorter than forensic investigations, warranting a smaller t_s. Considering n_A, we posit the default $n_A = 100$. This value is a trade-off between having enough actors to capture nuances in the insight gaining process and offset the stochastic generation process on the one hand, and not having too many so that the evaluation time stays reasonable. The third *AQ* parameter to be considered is the time it takes an actor to process one item (t^a_1). This simulates the time cost of processing individual items by a human analyst, which is certainly higher than if the machine would be allowed to iterate over results uninhibited. Indeed, even in rapid serial visual presentation (RSVP), users have been shown to only be able to perform basic processing (e.g., "is this an orange?") on 5–15 images per second, as shown by Van der

Corput and Van Wijk [34]. To the best of our knowledge, there is no decisive study on the "correct" value of t_1^a. For *AQ*, however, the main implication is that t_1^a remains the same across all methods. For simplicity, we posit the default $t_1^a = 1\ s$. If necessary for the domain of expertise, n_A and t_1^a can be both treated as a free parameter. Once the actors are generated and the session time is established, the evaluation itself can start.

Each evaluation session is first divided into *segments*, i.e., time periods between the insight changes by the actors when the actor's categories of relevance are constant. A subsession is ran for each segment, and each involves two parallel execution threads:

- *Actor thread* executes the artificial actor's interaction with the evaluated method.
- *Observer thread* records the progression of the actor.

Each session starts with the first query by the artificial actor and runs uninterrupted until t_s is reached. The time the evaluated method takes to produce the results is thus included in the session time.

The actor thread starts with the evaluated method suggesting items to the actor. Then, the actor indicates which items are relevant according to the current categories of relevance. An item is marked as relevant if it belongs to at least one category of relevance, and as not relevant otherwise. After processing each item, the actor sleeps for t_1^a seconds before proceeding further to simulate the time needed by a real user to process the item. Once the actor has given a complete relevance indication over all items suggested by a method, a time check is performed. If the elapsed time is greater or equal to t_s, the actor thread stops and records the set of all items ($\widehat{I}(t)$) and the set of all relevant items ($\widehat{I}_r(t)$) seen by the actor in the segment. Otherwise, the thread goes back to the first step, asking for new suggestions from the evaluated method.

The observer thread records the progression of the actor thread at certain equidistant time points further called *ticks*. The length of each tick, t_T, depends on t_s, and as such is also dependent on the domain and/or purpose of the evaluated method. At each tick t, the observer thread snapshots two sets: the set of all items seen by the user ($\widehat{I}(t)$), and the set of all relevant items seen by the user ($\widehat{I}_r(t)$). Note that the ticks start counting at the start of the session, rather than the start of each segment. The recorded values per ticks and segments are used to compute the results of *AQ*.

Once all sessions are complete, the *AQ* measures are computed. The measures, described in Section 3.4, take into account relevance, as well as other characteristics of multimedia analytic quality, namely the speed of the method, the diversity of the suggested items, and how well is the analyst able to estimate the percentage of relevant items in the collection. Finally, in Section 3.5, we explain how to interpret the results.

3.4 Evaluation measures

In this section, we describe the evaluation measures collected by *AQ*. Since *AQ* is a time-based evaluation method, each of the measures is a function of time. This allows for unbiased test of the analytic capabilities of the evaluated methods: measuring performance by recall steps does not reflect the computational efficiency of the evaluated method. *AQ* provides a clear overview of the methods' performance with respect to both relevance and efficiency.

The natural starting point is judging the relevance of the items suggested to the user by the evaluated method. For this, we use the classic metrics which have been the cornerstone of multimedia analysis evaluation for years: recall and precision. Let I_r be the set of relevant items in the collection and $\widehat{I}_r(t)$ the set of relevant items seen by the actor up until time t. I denotes the entire collection, and $\widehat{I}(t)$ corresponds to the set of items seen by the actor up until time t. Recall ($R(t)$) is defined as:

$$R(t) = \frac{|\widehat{I}_r(t)|}{|I_r|} \tag{6}$$

Precision ($P(t)$) is:

$$P(t) = \frac{|\widehat{I}_r(t)|}{|\widehat{I}(t)|} \tag{7}$$

These two metrics have time and again proved their mettle with respect to judging the relevance of the results. Another analytic consideration is the diversity of the results, as already considered by the diversity benchmarks and datasets mentioned in Section 2. An exploring analyst will want to see as many different kinds of items as possible. In general, it is difficult to evaluate diversity. Fortunately, for *AQ*, we can utilize the arbiter annotations. Let $\mathcal{A}$ denote the set of all arbiter annotations, and $\widehat{\mathcal{A}}(t)$ the set of annotations encountered in the seen items up until time t. The measure of diversity ($D(t)$) of results shown to the actor is then defined as:

$$D(t) = \frac{|\widehat{\mathcal{A}}(t)|}{|\mathcal{A}|} \tag{8}$$

A notion which is rarely taken into account by benchmarks is the evaluated method's speed with respect to producing the results, despite efficiency and responsiveness being a crucial requirement for many multimedia analytics systems. Since the evaluation sessions in *AQ* run uninterrupted, the time the evaluated method takes to produce results occupies an important portion of t_s and as such should be evaluated. Let $|\widehat{I}(t)|^{\max} = \lfloor \frac{t}{t_1^a} \rfloor$ denote the maximum number of items that the actor could have processed by time t, given the time to process 1 item (t_1^a). The throughput ($T(t)$) measure is then the ratio of the true number of items seen by the actor to the theoretical maximum:

$$T(t) = \frac{|\widehat{I}(t)|}{|\widehat{I}(t)|^{\max}} \tag{9}$$

The last important concern that we consider in this work is the ability to estimate the ratio of relevant items per category to the size of the entire collection. This ratio has been shown to greatly affect the perception of precision-recall results [8], so it is imperative that *AQ* reports it. Giving the user a correct impression about the ratio is also important for analytic purposes. For example, consider a medical scientist wanting to establish the percentage of patients with cancer. An analytic tool that she uses to explore the collection of body scans should be able to give her a correct estimate throughout the exploration. Let $RS_C = \frac{|I_C|}{|I|}$ denote the size of the set of the items in category $C \in C^a$ relative to the entire collection I and $\widehat{RS}_C(t) = \frac{|\widehat{I_C}(t)|}{|\hat{I}(t)|}$ denote the relative size estimate based on what the actor has seen up until time t. Using these values, we can enumerate how much the relative size estimate over- and underestimates the true value by computing $\frac{\widehat{RS}_C(t)}{RS_C}$ and $\frac{RS_C}{\widehat{RS}_C(t)}$, respectively. For the purposes of our measure, we treat over- and underestimation equally. The RELEVANCE PERCENTAGE ESTIMATE $(RPE(t))$ measure for actor a is then the minimum of these two values, averaged over all relevance categories of the actor:

$$RPE(t) = \frac{1}{|C^a|} \sum_{C \in C^a} \min\left(\frac{\widehat{RP}_C(t)}{RP_C}, \frac{RP_C}{\widehat{RP}_C(t)}\right) \tag{10}$$

For each of those measures, a value per each actor needs to be determined. Let $\bar{X}^a$ denote the final value of measure X for actor a. T and D are computed for the entire session at once by design, and hence $\bar{T}^a = T(t_s)$ and $\bar{D}^a = D(t_s)$. For R, P, and RPE, the value for actor a is the average over all segments seg in the actor's set of segments (denoted S_a):

$$\bar{X}^a = \frac{1}{|S_a|} \sum_{seg \in S_a} X(t_{seg}), \quad X \in \{R, P, RPE\} \tag{11}$$

Let $\bar{X}$ denote the final value of measure X. The final value for each measure $X \in \{R, P, D, T, RPE\}$ is then the average of values per actor:

$$\bar{X} = \frac{1}{n_A} \sum_{a \in A} \bar{X}^a, \quad X \in \{R, P, D, T, RPE\} \tag{12}$$

The analytic quality measure ($\mathbf{aq}_m$) of the evaluated method m is then the vector of the individual final values:

$$\mathbf{aq}_m = (\bar{R}, \bar{P}, \bar{D}, \bar{T}, \overline{RPE}) \tag{13}$$

3.5 Interpretation

Rather than being a single metric, AQ is a *collection* of metrics. Each of them captures a different aspect of the analytics process and can be easily interpreted. Apart from the classic presentation of the results in a table, an especially strong tool for interpreting results is the *time plot*, plotting time on the x axis and the individual measures on the y axis. For the time plots, individual values are computed per observer time tick, showing the development of the measures with respect to time with fine detail. The time frame provides a fair analytic comparison of individual methods by showing the naked truth about what the analyst sees and can work with at time t. The traditional recall-step-based plots, on the other hand, can easily disguise shortcomings with respect to key analytic aspects. For example throughput: as long as the method provides more relevant results, it is always shown to be the best, even if it is unusably slow. This can result in a warped view of the performance of the individual methods.

Comparing two sets of measures to assess ranking of multiple evaluated methods might be difficult. To tackle this problem, we propose AQ^{svm}, which learns the key aspects of the interplay between individual AQ measures. Let $\mathbf{aq}^a_m$ denote the vector of AQ measures corresponding to method m and actor a:

$$\mathbf{aq}^a_m = (\bar{R}^a, \bar{P}^a, \bar{D}^a, \bar{T}^a, \overline{RPE}^a) \tag{14}$$

Computing AQ^{svm}_m for method m then involves the following steps:

1. Collect AQ^a_{rand}, the set containing the AQ vectors for all actors $a \in A$ obtained by a random baseline, which always suggest a uniform random sample from the collection (without repetition):

$$AQ^a_{\text{rand}} = \bigcup_{a \in A} \mathbf{aq}^a_{\text{rand}} \tag{15}$$

2. Collect AQ^a_M, the set containing the AQ vectors for all actors $a \in A$ for all evaluated methods $m \in M$:

$$AQ^a_M = \bigcup_{\substack{m \in M \\ a \in A}} \mathbf{aq}^a_m \tag{16}$$

3. Train a linear SVM model using the vectors in AQ^a_M as positive training data and the vectors in AQ^a_{rand} as negative training data.

4. For each method m, AQ^{svm}_m is equal to the SVM score assigned to $\mathbf{aq}_m$ by the model trained in the previous step.

The intuition behind AQ^{svm} is as follows. The linear SVM model fits a separating hyperplane into the space of AQ measures such that the margin between positives (the actual evaluated methods) and the random baseline is maximal. We want our evaluated methods to be as far from this separating hyperplane (and thus, from the random baseline), as possible. Hence, comparing two methods, the one with higher AQ^{svm} wins. The difference between the AQ^{svm} values of the random baseline and the evaluated methods also indicate the benefit of using the methods over a simple random baseline: the higher the difference, the better. AQ^{svm} thus provides easy, powerful, and non-parametric aggregation of the AQ measures for easy comparison of methods.

Overall, using AQ measures has a number of distinct advantages. They provide rich information beyond simple relevance. Taken into account separately, they shed light on the evaluated methods' performance with respect to distinct key analytic aspects. Time plot analysis provides a fair comparison of methods, and it is easy to compare how the individual methods fare in different stages of the analytic session. AQ^{svm} provides a simple way to aggregate AQ measures for the purpose of ranking of individual methods. Moreover, individual AQ measures can be selected or discarded based on their importance to a specific purpose. This applies to all interpretation techniques presented in this section. Combining the time-based nature of the AQ measures with the insight-centered artificial actor model employed to obtain them yields a much broader and richer perspective of the evaluated methods' capabilities than the classic benchmark paradigm and shedding light on the potential insight gain by a real user.

4. SHOWCASE

In this section, we demonstrate the capabilities of AQ on an example analytic task. Note that AQ itself is independent of this showcase example, both in terms of task definition and the evaluated methods. For the showcase task, an

Table 1: Showcase AQ results.

Method	R	P	T	D	RPE
`random`	0.037	0.014	**0.999**	**0.443**	**0.405**
`vis`	**0.263**	0.134	0.532	0.218	0.105
`txt`	0.186	0.070	0.977	0.397	0.271
`mm`	0.203	**0.138**	0.465	0.239	0.108

Table 2: AQ^{svm} results for *search* (precision and recall) and *exploration* (precision and diversity).

Method	AQ^{svm}, *search*	AQ^{svm}, *exploration*
`random`	-0.11	0.12
`vis`	**1.73**	0.85
`txt`	1.02	0.45
`mm`	1.39	**0.87**

urban planner employed by the municipality of Amsterdam has downloaded a dataset of 20,000 Flickr images with their associated text related to Amsterdam. Her task is to quickly assess neighbourhood decay: deserted and/or poorly maintained buildings, graffiti, waste, loiterers etc. She wants to discover which neighbourhoods are problematic and what the problems are. The analytic session starts with the analyst providing three relevant examples of a multimedia item capturing a problematic aspect, i.e., with a query by example. Then, the session proceeds with the analyst going over the multimedia items provided by the analytic system, exploring the collection and interacting with the system throughout the session, steering the flow of the task by her current notion of relevance.

In our showcase, we want to see whether it is more beneficial for the analyst to use a system guiding her using a simple interactive learning algorithm, or if she is better off just selecting items at random. As designers, we want to see the build-up of quick insight, setting t_s to 15 minutes. For the other two parameters, number of actors (n_A) and time to process 1 item (t_1^a), we use the defaults, i.e., $n_A = 100$, $t_1^a = 1$ s. To evaluate the methods, we employ AQ. We are interested in the overall picture AQ paints. Since neighbourhood decay is a wide semantic notion, we will be looking at the performance of the individual methods with respect to using a broad image concept/topic dictionary. More specifically, we want to assess individual methods' capabilities with respect to both exploration and search, as the analyst in our case will need both: exploring the city and seeing diverse neighbourhoods, but also being able to search for specific signs of neighbourhood decay matching her relevance criteria.

The data processing step involved collecting the arbiter annotations and the visual and text features to be used for the interactive learning method. Since we are interested in the methods' performance on varying images, we opt for arbiter models that have been shown to work well on natural images and general text corpora. As the arbiter visual annotations, we used the 1000 ImageNet visual concepts provided by the Inception convolutional deep net conceived by Szegedy et al. [30], the winning entry of ILSRVC 2014. As the arbiter text annotations, we employed latent Dirichlet allocation to extract 100 latent topics using the Gensim framework [36]. As the visual features for the interactive learning method, we use 15k concepts provided by a custom Inception network retrained to provide annotations for all ImageNet classes with more than 200 examples. The text features for the interactive learning method are 300 latent topics again extracted using Gensim [36].

The random baseline (further denoted as `random`) simply returns a uniform random sample of the collection. The interactive learning method we use is a simple linear SVM model trained on the visual modality only (further denoted as `vis`), a linear SVM trained on the text modality only (`txt`); or two linear SVM models, one per modality, whose results are fused using Borda count (`mm`). The interactive learning method is seeded by three relevant examples provided by the user before the start of the session. After each relevance indication by the user, the interactive learning algorithm updates the positives and negatives, retrains the model accordingly, and provides its item suggestions based on the retrained model. All methods return 5 results at a time for the user to process. Each item can be seen up to one time, i.e., the user sees no repeated suggestions.

The AQ results are summarized in Table 1. If we would look at the results through the optics of the classic relevance-based paradigm, the clear winner is `vis`, winning over all competitors significantly in recall while being a very close second to `mm` on precision. All intelligent learning approaches dominate the `random` baseline on precision and recall. However, a broader look at all the components of AQ reveals that there are analytic components like diversity and relevance percentage estimation where `random` ranks first. These seem to be correlated to throughput, emphasizing that important elements of analytic quality can be dependent on the performance of the algorithm. In addition, `random`, due to it's fairness in selecting suggestions, covers a wider range of different kinds of items in the collection. These aspects of multimedia analytics are neither shown nor enumerated by the classic evaluation approaches.

To judge the exploration and search capabilities of individual methods, we obtain the respective AQ^{svm} scores. We aggregate precision and recall for the search AQ^{svm}, following the classic relevance-based paradigm. For the exploration AQ^{svm}, we use precision and diversity. The AQ^{svm} scores are summarized in Table 2. AQ^{svm} confirms the intuition with respect to search, which is purely relevance-based: `vis` comes out strongly on top, `mm` second, `txt` third. The `random` baseline is scored far from the intelligent learning methods, confirming the big gap with respect to relevance. In the exploration case, `mm` pulls slightly ahead of `vis`, which exactly reflects the situation when looking at D and P individually in Table 1. The gap between `random` and the winner is much smaller than in the case of search. This reflects that the `random` baseline is actually very strong in one of the characteristics and it is not as easy to discriminate `random` from the intelligent learning methods as in the case of search. AQ^{svm} is thus shown to provide a meaningful, comprehensive purpose-based ranking of methods.

The time plot of individual AQ metrics (Figure 4) gives additional insights with respect to the development of the metrics over time. Let $t = 150$ s be roughly the boundary of the early stage of exploration. Drawing this boundary, as shown in Figure 4, allows easy comparison of individual methods in the early vs. the late stage. In the precision plot, we notice that `mm` is strongly dominant in the early stage. After that point, the precision of `mm` and `vis` remains similar until the end of the session. This can be interpreted

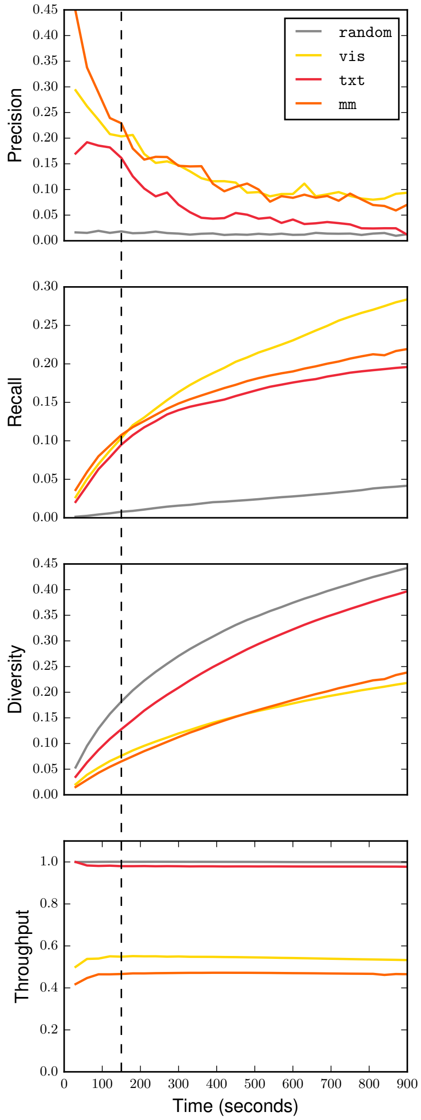

Figure 4: *AQ* time plots with $t = 150\ s$ marking the boundary of the early stage of the session.

as mm being very strong in the "pure query response" phase when the notion of relevance is constant. At the same time, it is lackluster in adapting to the insight changes, since it experiences the sharpest drop out of all intelligent learning methods. This gives a suggestion for improvement: since the model works well with a small number of positives/negatives and much worse with more, introducing decay to keep only the last few relevance indications might improve the performance. The recall plot confirms the modality synergy of the mm algorithm in the early stage and shows that mm is being significantly dragged down by the underperforming txt method in the later stages. The diversity plot reveals that the solid txt performance is not reflected in the mm method at all. The recall and diversity analyses give yet another suggestion: since the text modality is beneficial only initially, switching it off in the mm algorithm after a certain time might be considered to increase performance. As a side effect, this would increase throughput, which is also desirable. Overall, the time plot analysis reveals key strengths and weaknesses of individual algorithms, providing design insight into what to improve and what to exploit.

Compared with the classic evaluation taking into account only precision and recall or derivations thereof (mAP, F-measure etc.), *AQ* offers a much broader overview. In our case, it was able to characterize the four evaluated algorithms, pinpointing which techniques are more suitable for exploration and which for search. Moreover, the *AQ* analysis highlighted particular aspects to focus on in the case of mm method, which can be used to further improve its performance. This would not have been possible by just comparing individual values across evaluated methods, as is the case in the classic benchmark paradigm. *AQ* thus not only reveals *which* method is the best, but, unlike the classic evaluation approaches, also gives insights into the *why*.

5. CONCLUSION

In this paper, we presented *AQ* (analytic quality), a novel paradigm for multimedia analysis method design and evaluation. It significantly expands on the classic relevance evaluation paradigm by adding a number of real-life analytic considerations to the table, namely time, interactivity, and user insight. The time-based nature of *AQ* provides both the method designers and the end users with a picture about the analytic capabilities in a realistic time-limited setting, removing the strong assumption that the analyst will go over all the results. Including interactivity and evolving insight into the user model mimicks real user behaviour. This, in conjunction with using multiple evaluation metrics, each capturing a different aspect of the method, provides a much more detailed feedback and conclusions for further design improvement than just comparing two values of mean average precision. Another distinct advantage is the modularity of *AQ*. Metrics of interest can be selected for different method purposes, and new ones can be easily collected from the artificial actor sessions on demand. Moving multimedia analysis evaluation towards realistic analytic settings involving end users is certainly desirable. *AQ* is a leap forward on this path, bringing new challenges to the multimedia community which the computer vision and machine learning communities cannot yet address.

6. ACKNOWLEDGMENTS

This research is supported by the Dutch Technology Foundation STW, which is part of the Netherlands Organisation for Scientific Research (NWO), and which is partly funded by the Ministry of Economic Affairs.

7. REFERENCES

[1] D. Borth, R. Ji, T. Chen, T. Breuel, and S.-F. Chang. Large-scale visual sentiment ontology and detectors using adjective noun pairs. In *ACM MM*, pages 223–232, 2013.

[2] O. Chapelle, D. Metlzer, Y. Zhang, and P. Grinspan. Expected reciprocal rank for graded relevance. In *ACM CIKM*, pages 621–630, 2009.

[3] T. Demeester, D. Trieschnigg, D. Nguyen, K. Zhou, and D. Hiemstra. Overview of the TREC 2014 federated Web search track. In *TREC*, 2014.

[4] M. Eskevich, R. Aly, R. Ordelman, S. Chen, and G. J. F. Jones. The search and hyperlinking task at MediaEval 2013. In *MediaEval*, 2013.

[5] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010.

[6] T. M. Green, W. Ribarsky, and B. Fisher. Building and applying a human cognition model for visual analytics. *InfoVis*, 8(1):1–13, Mar. 2009.

[7] X.-S. Hua, Y. Ming, and J. Li. Mining knowledge from clicks: MSR-Bing image retrieval challenge. In *IEEE ICMEW*, pages 1–4, 2014.

[8] D. P. Huijsmans and N. Sebe. How to complete performance graphs in content-based image retrieval: Add generality and normalize scope. *IEEE TPAMI*, 27(2), Feb. 2005.

[9] B. Ionescu, M. Menéndez, H. Müller, and A. Popescu. Retrieving diverse social images at MediaEval 2013: Objectives, dataset, and evaluation. In *MediaEval*, 2013.

[10] B. Ionescu, A.-L. Radu, M. Menéndez, H. Müller, A. Popescu, and B. Loni. Div400: A social image retrieval result diversification dataset. In *ACM MMSys*, pages 29–34, 2014.

[11] A. Jaimes, N. Sebe, and D. Gatica-Perez. Human-centered computing: A multimedia perspective. In *ACM MM*, pages 855–864, 2006.

[12] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM TOIS*, 20(4):422–446, Oct. 2002.

[13] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM MM*, pages 675–678, 2014.

[14] M. A. Larson, B. Ionescu, X. Anguera, M. Eskevich, P. Korshunov, M. Schedl, M. Soleymani, G. Petkos, R. F. E. Sutcliffe, J. Choi, and G. J. F. Jones, editors. *Working Notes Proceedings of the MediaEval 2014 Workshop*, 2014.

[15] A. Lavie and A. Agarwal. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *StatMT*, pages 228–231.

[16] M. Lestari Paramita, M. Sanderson, and P. Clough. Diversity in photo retrieval: Overview of the ImageCLEF photo task 2009. In *Multilingual Information Access Evaluation II. Multimedia Experiments*, pages 45–59. 2010.

[17] Y. Li and B. Merialdo. VERT: Automatic evaluation of video summaries. In *ACM MM*, pages 851–854, 2010.

[18] C.-Y. Lin. ROUGE: A package for automatic evaluation of summaries. In *ACL*, pages 74–81, 2004.

[19] A. Nenkova, R. Passonneau, and K. McKeown. The pyramid method: Incorporating human content selection variation in summarization evaluation. *ACM TSLP*, 4(2), May 2007.

[20] C. North. Towards measuring visualization insight. *IEEE TCGA*, 26(3):6–9, 2006.

[21] C. North, P. Saraiya, and K. Duca. A comparison of benchmark task and insight evaluation methods for information visualization. *InfoVis*, 10(3):162–181, July 2011.

[22] P. Over, A. F. Smeaton, and G. Awad. The TRECVid 2008 BBC Rushes summarization evaluation. In *ACM TRECVid Video Summarization Workshop*, ACM MM, pages 1–20, 2008.

[23] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. BLEU: A method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002.

[24] W. A. Pike, J. Stasko, R. Chang, and T. A. O'Connell. The science of interaction. *InfoVis*, 8(4):263–274, 2009.

[25] S. Rudinac, M. Larson, and A. Hanjalic. Learning crowdsourced user preferences for visual summarization of image collections. *IEEE TMM*, 15(6), Oct. 2013.

[26] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *arXiv:1409.0575*, 2014.

[27] M. Sanderson, M. L. Paramita, P. Clough, and E. Kanoulas. Do user preferences and evaluation measures line up? In *ACM SIGIR*, pages 555–562, 2010.

[28] K. Schoeffmann. A user-centric media retrieval competition: The video browser showdown 2012-2014. *IEEE Multimedia*, 21(4):8–13, 2014.

[29] M. D. Smucker and C. L. A. Clarke. Time-based calibration of effectiveness measures. In *ACM SIGIR*, pages 95–104, 2012.

[30] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *arXiv:1409.4842*, 2014.

[31] J. J. Thomas and K. A. Cook. *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. IEEE Computer Society, 2005.

[32] B. Thomée and M. S. Lew. Interactive search in image retrieval: a survey. *Int J of MIR*, 1(2):71–86, July 2012.

[33] S. Tong and E. Chang. Support vector machine active learning for image retrieval. In *ACM MM*, pages 107–118, 2001.

[34] P. van der Corput and J. J. van Wijk. Effects of presentation mode and pace control on performance in image classification. *IEEE TVCG*, 20(12):2301–2309, Dec. 2014.

[35] R. Vedantam, C. L. Zitnick, and D. Parikh. CIDEr: Consensus-based image description evaluation. *arXiv:1411.5726*, 2015.

[36] R. Řehůřek and P. Sojka. Software framework for topic modelling with large corpora. In *LREC*, pages 45–50, 2010.

[37] J. Zahálka, S. Rudinac, and M. Worring. New Yorker Melange: Interactive brew of personalized venue recommendations. In *ACM MM*, pages 205–208, 2014.

[38] J. Zahálka and M. Worring. Towards interactive, intelligent, and integrated multimedia analytics. In *IEEE VAST*, pages 3–12, 2014.