Automated Component–Level Evaluation: Present and Future

Allan Hanbury[1] and Henning Müller[2]

[1] Information Retrieval Facility, Vienna, Austria
a.hanbury@ir-facility.org
[2] University of Applied Sciences Western Switzerland (HES–SO), Sierre, Switzerland
henning.mueller@sim.hcuge.ch

Abstract. Automated component–level evaluation of information retrieval (IR) is the main focus of this paper. We present a review of the current state of web–based and component–level evaluation. Based on these systems, propositions are made for a comprehensive framework for web service–based component–level IR system evaluation. The advantages of such an approach are considered, as well as the requirements for implementing it. Acceptance of such systems by researchers who develop components and systems is crucial for having an impact and requires that a clear benefit is demonstrated.

1 Introduction

Information retrieval (IR) has a strong tradition in evaluation, as exemplified by evaluation campaigns such as TREC (Text REtrieval Conference), CLEF (Cross Language Evaluation Forum) and NTCIR (NII Test Collection for IR Systems). The majority of IR evaluation campaigns today are based on the TREC organisation model [1], which is based on the Cranfield paradigm [2]. The TREC model consists of a yearly cycle in which participating groups are sent data and queries by the organisers, and subsequently submit retrieval results obtained by their system for evaluation. The evaluation produces a set of performance measures, quantifying how each participating group's system performed on the queries with a stable data set and clear tasks that evaluate the entire system.

This approach has a number of disadvantages [3]. These include:

- fixed timelines and cyclic nature of events;
- evaluation at system–level only;
- difficulty in comparing systems and elucidating reasons for their performance.

These disadvantages are discussed in more detail in Section 2. To overcome these disadvantages, we suggest moving towards a web–based component–level evaluation model, which has the potential to be used outside of the cycle of evaluation campaigns. Section 3 discusses some existing approaches to web–based and component–level evaluation, with examples of systems and evaluation campaigns adopting these approaches. Section 4 presents the promising idea of using

M. Agosti et al. (Eds.): CLEF 2010, LNCS 6360, pp. 124–135, 2010.

a web service approach for component–level evaluation. As use by researchers of the existing systems is often lacking, we pay particular attention to motivating participants in Section 5. Long term considerations are discussed in Section 6.

2 Disadvantages of Current Evaluation Approaches

This section expands on the disadvantages listed in the introduction. The first disadvantage is the *cyclic nature of events*, with a fixed deadline for submitting runs and a period of time during which the runs are evaluated before the evaluation results are released. There is usually also a limit on the number of runs that can be submitted per participant, to avoid an excessive workload for the organisers. At the end of each cycle, the data, queries and relevance judgements are usually made available to permit further "offline" evaluation. However, evaluating a system on a large number of test datasets still involves much effort on the part of the experimenter. A solution that has been proposed is online evaluation of systems, as implemented in the EvaluatIR system[1] [4]. This system makes available testsets for download, and allows runs in TREC runfile format (trec_eval) to be uploaded (they are private when uploaded, but can be shared with other users). It maintains a database of past runs submitted to TREC, benchmarks of IR systems and uploaded runs that have been shared, and supports a number of methods for comparing runs. This system not only allows evaluation to be performed when the user requires it, but it makes it possible to keep track of the state–of–the–art results on various datasets.

A further disadvantage is the *evaluation at system–level only*. An IR system contains many components (e.g. stemmer, tokeniser, feature extractor, indexer), but it is difficult to judge the effect of each component on the final result returned for a query. However, extrapolating the effect on a complete IR system from an evaluation of a single component is impossible. As pointed out by Robertson [5], to choose the optimal component for a task in an IR system, alternatives for this component should be evaluated while keeping all other components constant. However, this does not take into account that interactions between components can also affect the retrieval results. For research groups who are experts on one particular component of an IR system, the requirement to evaluate a full system could mean that their component is never fully appreciated, as they do not have the expertise to get a full IR system including their component to perform well.

A further difficulty due to the system–level approach is that when reviewing a number of years of an evaluation task, *it is often difficult to go beyond superficial conclusions based on complete system performance and textual descriptions of the systems*. Little information on where to concentrate effort so as to best improve results can be obtained. Another possible pitfall of the system–level approach, where the result of an evaluation is a ranked list of participants, is the potential to view the evaluation as a competition. This can lead to a focus on tuning existing systems to the evaluation tasks, rather than the scientific goal of determining how and why systems perform as they do. Competitions generally favor small

[1] `http://www.evaluatir.org/`

optimizations of old techniques rather than tests with totally new approaches with a possibly higher potential.

3 Review of Web–Based and Component–Level Evaluation

Several existing campaigns have already worked with component–level evaluation or at least an automated approach to comparing systems based on a service–oriented architecture. The approaches taken are listed below, and each of them is then discussed in more detail:

1. Experimental framework available for download (e.g. MediaMill);
2. Centralised computer for uploading components (e.g. MIREX);
3. Running components locally and uploading intermediate output (e.g. NTCIR ALCIA, Grid@CLEF);
4. Communication via the web and XML based commands (e.g. [6]);
5. Programs to be evaluated are made available by participants as web services (e.g. BioCreative II.5 online evaluation campaign).

For the first approach listed, an experimental framework in the form of a search engine designed in a modular way is made available for download. It can be installed on a local machine, and the modular design should allow simple replacement of various components in the installed system provided that a number of design parameters are satisfied. An example from a related domain is the MediaMill Challenge [7] in the area of video semantic concept detection. A concept detection system, data and ground truth are provided, where the system is broken down into feature extraction, fusion and machine learning components, as shown in Figure 1. Researchers can replace any of these components with their own components to test the effect on the final results. An advantage of this approach is that all processing is done on the local machine. A disadvantage is that the framework always represents a baseline system, so improvements are compared to a baseline, not the state–of–the–art. A solution could be an online repository, similar to EvaluatIR mentioned above, which allows results but also components to be shared. Having a system with a fixed number of components and strict workflow also has the problem that new approaches must fit the workflow, limiting the freedom to implement radically new ideas.

The second approach involves making available a server onto which components can be uploaded and incorporated into an evaluation framework running on the server. The components would again have to satisfy various design parameters. A simple way of doing this would be to provide contributors access to the server for uploading and compiling code, which can then be registered in the evaluation framework. Out of necessity due to the copyright restrictions on distributing music, this approach has been adopted by the Music Information Retrieval Evaluation Exchange (MIREX) [8] — one copy of the data is kept on a central server, and participants submit their code through a web interface to be run on the data. The advantage is that the data and code are on the same

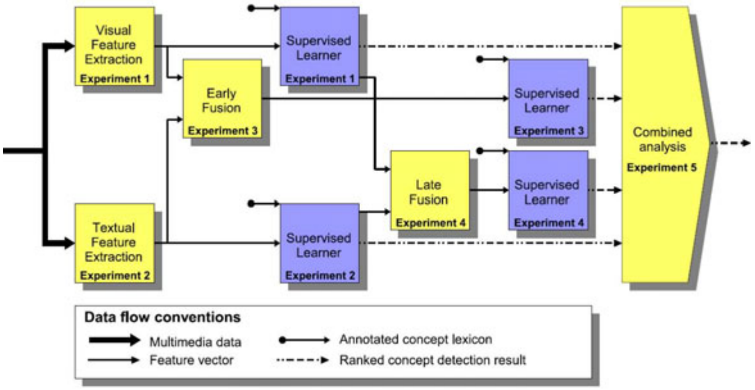

Fig. 1. The concept detection system diagram for the MediaMill Challenge (from [7])

server, so the distribution of huge datasets is avoided, and the evaluation experiments can be run efficiently. The disadvantage for the participants could lie in getting their code to compile and run on a remote server and the risks associated with missing libraries, etc. Participants, in particular companies, could object to uploading proprietary code to a foreign server. Such a system also creates large overheads for the organiser — for MIREX, managing and monitoring the algorithms submitted consumes nearly 1000 person–hours in a year [8].

The Grid@CLEF initiative[2] represents the approach of running components locally and uploading intermediate output. It implemented a component–level evaluation as a pilot track in CLEF 2009 [9]. In order to run these experiments, the *Coordinated Information Retrieval Components Orchestration (CIRCO)* [10] framework was set up. A basic linear framework consisting of tokeniser, stop word remover, stemmer and indexer components was specified (Figure 2). Each component used as input and output XML data in a specified format, the CIRCO Schema. An online system (CIRCO Web) was set up to manage the registration of components, their description and the exchange of XML messages. This design is an intermediate step between traditional evaluation methodologies and a component–based evaluation — participants run their own experiments, but are required to submit intermediate output from each component. The advantages, as pointed out in [9], are that the system meets the component–level evaluation requirements by allowing participants to evaluate components without having to integrate the components into a running IR system or using an API (Application Programming Interface) due to the XML exchange format. This also allows the components to be evaluated asynchonously, although to an extent limited by the necessity of having output from early components available before being able to test later components in the sequence. The disadvantage pointed out in [9] is that the XML files produced could be 50–60 times the size of the original collection, making the task challenging from a computational point of view.

[2] `http://ims.dei.unipd.it/websites/gridclef/`

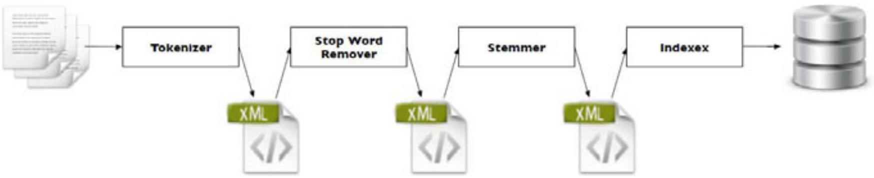

Fig. 2. The Grid@CLEF approach to component–level evaluation (from [9])

A similar approach is adopted in the NTCIR–7 and NTCIR–8 Advanced Cross–lingual Information Access (ALCIA) task[3]. By using the specified XML format, the output from IR modules can be used as input for the Question Answering (QA) modules. It is interesting to note in the NTCIR–7 results [11] that combinations of IR and QA modules from different groups always outperformed combinations from a single group.

In similar ways ImageCLEF has created a nested approach where the output of a first step is distributed to all participants as an input to further steps. In general, textual retrieval results and visual retrieval results are made available to all participants, as not all participants work in both domains. Another step was taken with the Visual Concept Detection Task (VCDT) in 2008, where the results of this task — information on concepts detected in images, such as sky, buildings, animals, etc. — were made available to participants of the photo retrieval task, which used exactly same data set [12]. This gave the participants another source of (potentially noisy) information with which to augment their systems. Unfortunately, few groups integrated this additional information, but those who did showed improved results as an outcome. In the medical task of ImageCLEF 2010 a modality classification task for the entire data set is added before the retrieval phase and results will be made available to participants as for the VCDT task above. Past experiments of the organizers showed that adding modality information to filter the results improved all submitted runs.

An idea for fully automatic evaluation has already been proposed for image retrieval in 2001 [13]. The communication framework (MRML — Multimedia Retrieval Markup Language) was specified, and a web server for running the evaluation by communicating in MRML over a specified port was provided. This system unfortunately did not receive much use as the implementation of a MRML framework would result in additional work for the researchers. The framework also had the disadvantage that the database was fixed (it could of course be extended to several data sets); that due to little use there were few comparisons with state of the art techniques; and that mainly the GNU Image Finding Tool (GIFT[4], [14]) that implements MRML natively was the baseline.

An example of the use of web services in evaluation from a related domain is the BioCreAtIvE (Critical Assessment of Information Extraction systems in Biology) Challenge for annotation in the biomedical domain [15,16]. The approach

[3] http://aclia.lti.cs.cmu.edu/ntcir8/

[4] http://www.gnu.org/software/gift/

for BioCreative II.5[5] was to have all participants install a web server to make their Annotation Servers available. A central MetaServer then calls the available Annotation Servers that use a standard interface. The Annotation Servers take as input a full–text document and produce as output the annotation results as structured data. The advantage of such a system is that researchers can maintain their tools locally and thus do not need to concern themselves with installation on another machine. Furthermore, the resources run in a distributed way thus limiting the charge on a central server. The system also allows the evaluation of efficiency of the tools, such as the response speed, which is an important criterion when working with extremely large databases. On the other hand such a system can favor groups with large hardware budgets.

Table 1 summarises the advantages and disadvantages of the approaches.

4 Towards Web–Based Component–Level Evaluation

To overcome the disadvantages of system–level evaluation, it is necessary to move IR evaluation towards web–based component–level evaluation. In the BioCreative challenge, complete systems are made available as web services. This is an approach that would also work in the IR evaluation framework, where participants could expose their search engines through web services. Queries (or finer–grained tasks) could be sent to these web services by the central Metaserver, and document lists sent back to the Metaserver for further evaluation. The database of documents to index could be provided as a download as is usual in evaluation campaigns, and could later be developed so that documents to index are provided by the Metaserver. This web service–based IR evaluation opens the door to a component–level evaluation built on the same principles.

A schematic diagram of a general component–based evaluation system is shown in Figure 3. The basic idea is that a framework consisting of components (an arbitrary framework diagram is shown in Figure 3) is defined, and contributors can add instances of the components into this framework for use in running IR experiments. The main challenge is how to instantiate such a framework so that researchers can easily add components to it, and experiments can be successfully run without creating much additional work for researchers.

The general techniques for developing automatic evaluation systems exist [6] with the Internet and service–oriented architectures, for example. If all researchers made their components available via a standardized interface, then these components could all be shared by the various participants and used in many combinations. Already now many IR tools are based on existing components such as Lucene, Lemur or other open source projects and having service–oriented use of such tools would simply be one step further. This could even help researchers to better integrate existing tools and techniques.

Such an evaluation would work as follows. An IR system built out of a set of components will be specified. Participating groups in the evaluation may choose which components they wish to submit and which components to use from other

[5] `http://www.biocreative.org/events/biocreative-ii5/biocreative-ii5/`

Table 1. Comparison of the current approaches to component–level evaluation

Approach	Example	Advantages	Disadvantages
Experimental framework download	Mediamill	– Processing done locally	– Framework always represents a baseline system – Fixed workflow
Centralized component upload	MIREX	– Distribution of huge datasets avoided – Evaluation experiments can be run efficiently	– Getting code to compile and run on a remote server – Large overheads for the organisers
Uploading intermediate output	Grid@CLEF, NTCIR ALCIA	– Participants evaluate components without integrating them into complete running systems – Components can be evaluated asynchronously	– Asynchronous evaluation restricted by pipeline structure – XML files produced 50–60 times the size of the document collection
Communication via the web and XML based commands	MRML	– fully automatic and quick evaluation at any time	– Overhead through implementation of the MRML framework – Databases fixed
Web services	BioCreative II.5	– Tools can be maintained locally – Programs run in a distributed way	– Can favour groups with a large hardware budget – Overhead of writing a web service interface to a system

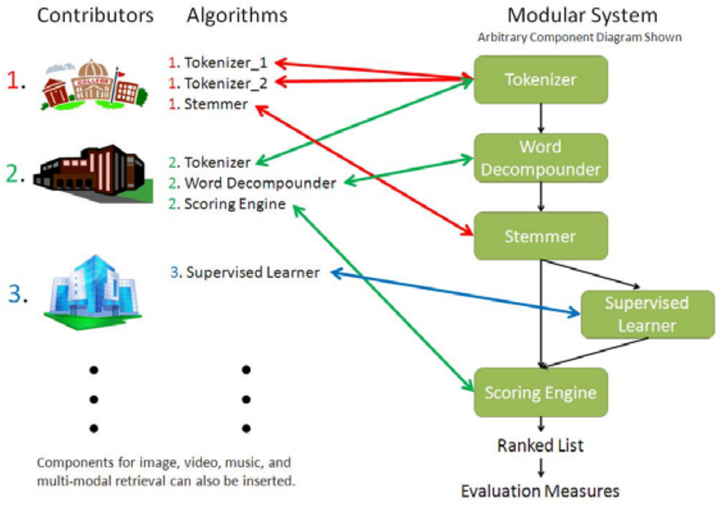

Fig. 3. Overview of a general component–level evaluation framework

groups. These components should be written so as to run on the participants' computers, callable through a web interface. Participants register their components on a central server. The central server then runs the experiments using a large number of combinations of components, accessed through their web interfaces. To create such a system, the following are needed:

- software and a central server to run the evaluation;
- protocols for interfacing with programs over the web, exchanging data and exchanging results, with the current standard for this being web services, so XML–based protocols;
- as for any IR evaluation: large amounts of data, realistic queries and relevance judgements.

The protocol design is the key challenge. The participants' task will shift from performing the experiments to adapting their code to conform to the protocols. In order to make this attractive to participants, the protocols should be designed to have the following properties:

Stability: The protocols should be comprehensively designed to change little over time — after an initial effort to get their systems compliant, little further *interface work* would have to be done by the participants (a standard really needs to be created).

Simplicity: The initial effort by participants to get their systems compliant should not be high, as a large initial hurdle could discourage participation. In addition to a specification, code implementing key interface components should be provided.

Wide Applicability: Implementing the protocols should enable groups to achieve more than participation in a single evaluation campaign. Standardizing the protocols for different evaluation campaigns and potentially for other uses is therefore important.

These properties can be contradictory. A stable protocol that covers all possible eventualities, anticipating all current and future needs, is less simple. A complex protocol could be made simpler by having many optional elements (e.g. the output of a tokeniser could be just a stream of words, or it could include information on word position, sentence boundaries, etc.). With such optional elements, downstream components would require the ability to handle missing elements, or to specify which elements are required so that they can function. Wide applicability can be obtained through the use of a common web service protocol, however many of these protocols do not meet the requirement for simplicity.

For the control software, as the amount of participation increases and the number of components included in the IR system specification increases, the potential number of component combinations will explode. It will therefore not be feasible to test all possible combinations. Algorithms for selecting potentially good component combinations based on previous experimental results and the processing speeds of components, but with low probability of missing good combinations, will have to be designed. It would also be useful for users to have available baseline components with "standard" output to simplify the integration of a new component. Further difficulties to be considered are the remote processing of large amounts of data, where participants with slower Internet connections may be disadvantaged (an initial solution may be to continue distributing the data to be installed locally). It will also be good to ensure that participants with less computing capacity are not at a large disadvantage.

A current problem in IR evaluation that is not addressed at all in this framework is the provision of sufficient data, queries and relevance judgements. With the potential for more efficient experiments, this problem might become worse.

5 Motivating Participation

It is important to design the system so that it is accepted and used by the targeted researchers. The system should be designed so that there are clear benefits to be obtained by using it, even though an initial effort is required to adopt it. The component–level evaluation approach has the following general advantages and benefits:

- A large number of experiments can be executed. Each participant makes available online components, which are then called from a central server. This reduces the amount of work for each participant in running complete IR experiments and allows to reuse components of other participants. More extensive experimental results on component performance can be obtained.
- The best performing combination(s) of components can be identified, where components making up this best performing combination could be from different groups. Different search tasks will also possibly be best performed by different constellations of components allowing even for a query–specific optimization of techniques.

- Significantly less emphasis will be placed on the final ranking of complete systems. The results will be in the form of constellations of which components are best suited for which tasks. It also reduces the perceived competitiveness by removing the ranked list of participants. On the other hand, it is easier for a researcher to have his/her component “win” as there are several categories and not only a best final system.
- Research groups will have the opportunity to concentrate on research and development of those components matching their expertise.
- The reuse of components by other researchers is facilitated. By having other research groups’ components available, the building of systems can become easier and other systems using components can increase the number of citations received by publications describing these components.

Despite these general advantages and benefits, there is currently a very low acceptance of component–level evaluation among researchers. For MediaMill, browsing the papers citing [7] gives the idea that while many researchers make use of data and ground truth, few use the system framework. The MRML–based system basically had two users, and there were also only two participants in Grid@CLEF 2009. BioCreative II.5 on the other hand had 15 groups participating showing that such an integration is possible.

When introducing component–level evaluation, the benefits should be made clear through a publicity campaign as a critical mass of participants needs to be reached. It is expected that web service–based systems will become common and thus many researchers might have an interest in such an interface anyway. The main users of such systems will most likely be PhD students and post–docs and for them this can be a much easier start through having a clear framework and not losing time working on already existing parts in poorer quality.

6 Long–Term Considerations and Conclusions

Benchmarking and technology comparisons will remain an important domain to advance science and particularly a domain where good baselines exist and where the application potential is already visible. Such benchmarking has to become more systematic and has to be finer–grained than is currently the case. A compromise also needs to be found leaving researchers the possibility to have totally new approaches but at the same time allowing existing components to be reused. This should make the entire process more efficient and allow researchers to concentrate on the novel parts. Particularly PhD students could benefit extensively from such a concept as the entry burden would be much lower than at present.

Given the additional experimental data that will become available through such a framework, a long–term aim can be to design a search engine that can be built entirely from components based on the task that a user is carrying out and analysis of his/her behaviour (targeted search, browsing, etc.). The ability to clearly see the effect of changes in components on the results of a system should also contribute to solving the problem described in [4]: it is not clear from results in published papers that IR systems have improved over the last decade. The more components that can be called the better the acceptance of such a system will be. A possible implementation of such complex IR systems could be through a workflow paradigm, following the lead of eScience with systems such as Kepler[6] [17]. It might be beneficial to have a centrally managed infrastructure where components can be made available also from groups that lack the computing power to host components. Workflow systems also work in a Grid/Cloud environment [18], which could address the large-scale requirements of a component-based IR system.

There is a large number of challenges that need to be tackled. The problem of obtaining a sufficient number of queries and relevance judgements to allow large scale experiments has to be considered. Innovative approaches to harnessing Internet users for continuously increasing the number of relevance judgements should be examined, such as games with a purpose [19] or remunerated tasks [20]. Furthermore, extremely large databases have now become available but are still only rarely treated by researchers. Another problem is changing databases in which documents are constantly being added and deleted, e.g. FlickR.

A possible first step towards automated component–level evaluation is to create a full system approach for IR evaluation (as in Biocreative II.5). For simplicity, the data should be sent to participants and installed locally as is currently done in evaluation campaigns. Each participant should create a web service interface to their full search system, which can be called by the central server. This will allow research groups to get practice at using the web service approach. Once this approach has been accepted by the research community, the component–level evaluation can be introduced in a stepwise way.

Acknowledgements

This work was partially supported by the European Commission FP7 Network of Excellence PROMISE (258191). We thank the referees who provided a number of excellent suggestions.

References

1. Harman, D.: Overview of the first Text REtrieval Conference (TREC–1). In: Proceedings of the first Text REtrieval Conference (TREC–1), Washington DC, USA, pp. 1–20 (1992)
2. Cleverdon, C.W.: Report on the testing and analysis of an investigation into the comparative efficiency of indexing systems. Technical report, Aslib Cranfield Research Project, Cranfield, USA (1962)
3. Robertson, S.: On the history of evaluation in IR. Journal of Information Science 34, 439–456 (2008)
4. Armstrong, T.G., Moffat, A., Webber, W., Zobel, J.: Improvements that don't add up: ad-hoc retrieval results since 1998. In: CIKM 2009: Proceeding of the 18th ACM Conference on Information and Knowledge Management, pp. 601–610. ACM, New York (2009)

[6] `http://kepler-project.org/`

5. Robertson, S.E.: The methodology of information retrieval experiment. In: Jones, K.S. (ed.) Information Retrieval Experiment, pp. 9–31. Butterworths (1981)
6. Müller, H., Müller, W., Marchand-Maillet, S., Squire, D.M., Pun, T.: A web–based evaluation system for content–based image retrieval. In: Proceedings of the 9th ACM International Conference on Multimedia (ACM MM 2001), Ottawa, Canada, pp. 50–54. ACM, New York (2001)
7. Snoek, C.G.M., Worring, M., van Gemert, J.C., Geusebroek, J.M., Smeulders, A.W.M.: The challenge problem for automated detection of 101 semantic concepts in multimedia. In: Proc. ACM Multimedia, pp. 421–430 (2006)
8. Downie, J.S.: The music information retrieval evaluation exchange (2005-2007): A window into music information retrieval research. Acoustical Science and Technology 29, 247–255 (2008)
9. Ferro, N., Harman, D.: CLEF 2009: Grid@CLEF pilot track overview. In: Working Notes of CLEF 2009 (2009)
10. Ferro, N.: Specification of the circo framework, version 0.10. Technical Report IMS.2009.CIRCO.0.10, Department of Information Engineering, University of Padua, Italy (2009)
11. Mitamura, T., Nyberg, E., Shima, H., Kato, T., Mori, T., Lin, C.Y., Song, R., Lin, C.J., Sakai, T., Ji, D., Kando, N.: Overview of the ntcir-7 aclia tasks: Advanced cross-lingual information access. In: Proceedings of the 7th NTCIR Workshop Meeting on Evaluation of Information Access Technologies (2008)
12. Deselaers, T., Hanbury, A.: The visual concept detection task in ImageCLEF 2008. In: Peters, C., Deselaers, T., Ferro, N., Gonzalo, J., Jones, G.J.F., Kurimo, M., Mandl, T., Peñas, A., Petras, V. (eds.) CLEF 2008. LNCS, vol. 5706, pp. 531–538. Springer, Heidelberg (2009)
13. Müller, H., Müller, W., Marchand-Maillet, S., Pun, T., Squire, D.: A web-based evaluation system for CBIR. In: Proc. ACM Multimedia, pp. 50–54 (2001)
14. Squire, D.M., Müller, W., Müller, H., Pun, T.: Content–based query of image databases: inspirations from text retrieval. In: Ersboll, B.K., Johansen, P. (eds.) Pattern Recognition Letters (Selected Papers from The 11th Scandinavian Conference on Image Analysis SCIA 1999), vol. 21, pp. 1193–1198 (2000)
15. Krallinger, M., Morgan, A., Smith, L., Leitner, F., Tanabe, L., Wilbur, J., Hirschman, L., Valencia, A.: Evaluation of text-mining systems for biology: overview of the second biocreative community challenge. Genome Biology 9 (2008)
16. Morga, A.A., Lu, Z., Wang, X., Cohen, A.M., Fluck, J., Ruch, P., Divoli, A., Fundel, K., Leaman, R., Hakenberg, J., Sun, C., Liu, H., Torres, R., Krauthammer, M., Lau, W.W., Liu, H., Hsu, C., Schuemi, M., Cohen, K.B., Hitschmann, L.: Overview of BioCreative II gene normalization. Gene Biology 9, S2–S3 (2008)
17. Ludäscher, B., Altintas, I., Berkley, C., Higgins, D., Jaeger, E., Jones, M., Lee, E.A., Tao, J., Zhao, Y.: Scientific workflow management and the Kepler system. Concurrency and Computation: Practice and Experience 18, 1039–1065 (2006)
18. Wang, J., Crawl, D., Altintas, I.: Kepler + Hadoop: a general architecture facilitating data-intensive applications in scientific workflow systems. In: WORKS 2009: Proceedings of the 4th Workshop on Workflows in Support of Large-Scale Scienc, pp. 1–8. ACM, New York (2009)
19. von Ahn, L.: Games with a purpose. IEEE Computer Magazine, 96–98 (2006)
20. Alonso, O., Rose, D.E., Stewart, B.: Crowdsourcing for relevance evaluation. SIGIR Forum 42, 9–15 (2008)