Practical Issues in Information Access System Evaluation

Evangelos Kanoulas
University of Amsterdam
e.kanoulas@uva.nl

Jussi Karlgren
KTH Royal Institute of Technology and Gavagai, Stockholm
jussi@kth.se

May 3, 2017

1 Initial motivations

The last six decades information retrieval scholars and practitioners have put an enormous effort to develop an arsenal of evaluation techniques and frameworks that vary from collection-based approaches to user studies and in-situ methods to assess the effectiveness of information access algorithms, technologies, and systems. Most of these methods take adhoc search as the use case to address. Simultaneously, practical information access services are now embedded in many other commercial services and products, and many new services have been introduced, many of which incorporate search or classification as central components but with use cases that range beyond search. These components are seldom evaluated rigorously, and if they are, the results of those evaluation exercises are seldom perceived as relevant for system design in an industrial setting.

Small and medium size enterprises often lack the resources needed to develop proper evaluation infrastructures, but also to follow the research development in the field of evaluation. Similarly, academics lag behind in (a) understanding real practical issues raised when it comes to the evaluation of real systems - e.g. even depth-k pooling is often infeasible when an SME has a single ranking algorithm developed, and (b) sensing the breadth of applications and tasks on which systems require evaluation and the challenges of them. Large enterprises with the necessary resources and the data sets and flows to work with are hesitant to make their tests public, for both commercial and legal reasons.

This workshop brought together representatives from technology companies, large and small, media houses, industrial consultants and academic research in information access for a discussion on practical issues and solutions to these issues. It took the industrial day panel from SIGIR 2016 as a partial inspiration for further discussions.

Table 1: Workshop agenda.

09:00	Tea and coffee
09:30	Welcome
09:45	Presentation round
10:00	Challenge question review
11:00	Witness statements
12:30	Lunch
13:30	Discussion, in groups
15:30	Tea and coffee
16:00	Outcome formulation
16:30	Brief discussion on reporting
17:00	Closing
19:00	Dinner

2 Workshop details

The workshop was attended by 26 people from 25 organisations, and 5 countries, graciously hosted by the British Computing Society and its Information Retrieval Specialist Group in conjunction with the annual Search Solutions event.

Prior to the workshop participants were asked to think of some concrete cases where some form of evaluation exercise (whether academic or industrial) proved to be or not to be useful, some concrete cases where some form of evaluation would really be of use if one were available, or some case where an academic evaluation effort would be in need of validation of its metrics. Participants were provided with a provisional list of "challenge" questions, to think about. Below is the set of questions the workshop attempted to answer:

- Do you do systematic evaluation for your work at present? Do you find that it is useful? Do your colleagues find it useful?
- What actions does an evaluation precipitate?
- Do your evaluation efforts relate to quality metrics of other types, key performance indicators, downstream satisfaction metrics, sales, etc?
- Do you make a distinction between satisfying or optimising a process?
- How could we address the difference between cutting edge and state of the art?

The workshop took the format of a round-table discussion. The agenda can be found in Table 1. Participants introduced themselves and made an opening statement identifying the most significant evaluation issue they face in their organization or encounter in their work. The issues raised were clustered around end-to-end versus system component evaluation, user-oriented evaluation under different conditions and environments, (e.g. user satisfaction in mobile apps, user satisfaction when there is no ability to collect any user feedback, user-oriented measures beyond satisfaction such as engagement). Customer satisfaction (different from the end-user satisfaction)

was also raised as an important issue since different customers have different needs, there is a long tail of customer, and it is not always straight-forward how to translate what the customer wants from an information access system to an evaluation measurement.

The second round further elaborated on the issues raised and led to breakout groups which focused on 4 practical issues that came out as significant: (a) how to promote evaluation within an organization, (b) how to evaluate individual components of an information access systems and how is this related with an end-to-end evaluation, (c) what metrics should one use under different conditions, and (d) how to evaluate dynamic & interactive systems for which there is no benchmarking available. The workshop ended with an effort from the participants to formulate conclusive recommendations to industrial and academic organizations.

At the end of the working day, the workshop participants recongregated for a somewhat less formal discussion in the nearby Covent Garden.

3 Insightful observations

The participants brought diverse experiences and observations to the discussion.

First and foremost, the discussion circled around the question of what to evaluate — an entire product and processing pipeline or single components? The downstream output of a larger system is arguably most important to evaluate, but, as some pointed out, that is evaluated through sales and customer satisfaction anyway and any other evaluation is redundant. End-to-end evaluation, it was also pointed out, obscures the performance of components. As a general returning point, the view that testing an integrated search engine component in a larger system is very similar to unit testing and should be introduced into the same test scheme as any other component. Unit testing (or a similar activity) translates well into development or tuning action, whereas end-to-end testing does not. However, the pass/fail-nature of most unit tests are different from the more incremental relevance tuning efforts, however, and the question is then if arbitrary thresholds for acceptability should be accepted or recommended and how those should be set.

Secondly, the target notion for evaluation was discussed in some detail, without a clear resolution of the discussion. To a certain extent the quality of a search engine is a known dimension of variation, but every system choice is a three-way (at least) optimisation of (i) quality; (ii) time, cost, and attention; (iii) response speed and system performance.

Thirdly, the relevance of a benchmarking scheme, it was noted, hinges on the validity of the use case that benchmark is built upon. Unless the target notion extracted from that benchmark lacks in validity for the business logic of the organisation in question, the validity of entire test will be low.

Fourthly, the vast majority of industrial installations have no overview of the various components they consist of. "I have 50 web sites to manage, and I have no idea how they are doing", said one industrial participant. Another claimed that evaluation is a luxury, since "everything out there is broken" and needs to be fixed first.

Fifthly, new projects and new development ideas which often drive information technology in an organisation often introduce complexity on top of existing functionality. Attention shifts to new functions and the older components are assumed to be stable. This is not always true. Also, the process of introducing new ideas includes feasibility decisions ("will this fly"?) which is different from improvement measurements ("will it handle more clicks?").

Bringing the above points together, is of course the general scarcity of resources and attention: raising evaluation to higher priority, it was suggested, would be impossible on its merits alone, but needs to be inserted into existing workflows and processes, in ways so as not to incur more effort. Related to this, which to a large extent is an organisational issue, many participants stressed the need for using soft skills, understanding the client organisation and its business processes and internal logic, and the necessity to "whitecoat" some recommendations using academic authority and reference to best practice and established research. Having this in mind results in evaluation expertise veering very close to general business consultancy.

Scarcity was also discussed in relation to data resources. The discussion evolved around the reusability of experimental data, how to properly collect data by injecting randomization into your algorithm that can help evaluate future algorithmic developments. The difficulty of collecting reusable data was also related to UI experimentation and development. Time was also spent on discussing academic benchmarking. A number of participants pointed out that most of the academic datasets are not open, often not representative of the data their systems opperate on, and are lacking the variability in user and customer needs experience in industry. A remark however was made that often the performance trends on academic and industrial datasets are strongly correlated.

A recurring theme in every discussion on repeatable and reproducible experimentation is that of shareable data sets and the intellectual property rights attached to those data sets. An argument was proposed that the most important aspect of a test is to indicate if a component behaves predictably under certain conditions, not necessarily identical but comparable to some known set of conditions, and that thus, sharing data sets might not be necessary even in academic research. In industrial settings, in any case, proprietary data sets would most likely be the norm.

A further question brought up by some is whether formal certifications such as ISO 9000 and similar would be beneficial for audit purposes. The opinion as to this was divided among participants.

4 Demanding challenges

In the discussion, the challenges for systematic evaluation can be grouped in to three rough categories: organisational, technical, and individual issues.

4.1 Organisational issues

Evaluation of quality of output, especially of embedded components, is seldom at the forefront of organisational attention. As observed by a participant with a consultancy background, customers tend to purchase only one IT solution, with relatively little attention to quality as compared to attention to how well its features fit with the existing organisational use case. The people who need to make decisions on issues regarding IT quality, it was observed, frequently do not know who they are. Quality issues mostly are observed by sales staff, not technology departments, and if customer votes with feet rather than requesting support, if quality comes up in discussion with sales staff, that feedback as well as NPS metrics or churn rate assessments are rarely passed back to development staff.

4.2 Technical issues

Many known technical issues make systematic evaluation practices cumbersome in an industrial setting. Lack of data sets without complex ownership issues, lack of established benchmark figures and resistance to sharing benchmark numbers due to real or imagined commercial harm of doing so, and lack of system overview stand in way of academic style evaluation practices. In system development processes, new features often are evaluated using whatever metrics can be found, but those are typically not retained when a component is built and scaled up to production and later deployed. The understanding of evaluation is that it is a decision-making criterion, rather than a monitoring tool.

4.3 Mindset issues

Quality improvement issues are routinely relegated to the back burner, ranking lower than bug fixes or feature introduction. The debate in software engineering between prioritisation between bugs and features is lively and productive, but quality assurance is rarely mentioned in that context. The major question here is between short-term vs long-term optimisation of system function, and while hardware issues and software operations routinely take long-term views, in questions such as redundancy, backups, maintainability, planned service outages etc, quality issues are often questioned and measured in terms of immediate hurt to customers or immediate output improvement. This scope needs to change for evaluation to be viewed with more interest by an organisation.

5 Useful principles

The workshop participants enunciated some basic principles to guarantee a certain level of evaluation becoming part of the culture in an organisation.

1. Use business logic and customer relationship as the key to motivating continuous quality assurance, and inject technical quality as part of that process.
2. Retain experimental culture in organisations — all new entrants have it from their education and the ethos of the IT field is based on it. Encourage and do not stifle it.
3. Stress internally in organisation that evaluation is the scaffolding for any information dense organisation, otherwise the organisation is flying blind.

6 Actionable recommendations

Finally, the workshop participants made efforts to formulate conclusive recommendations to industrial and academic organisations alike. Full agreement was reached only insomuch as this list cannot but be perceived as incomplete, provisional, and inspirational.

1. A recommendation from industry to academics: Stop chasing web search — it is just one very specific collection. Enterprise search and personal search are different and commercially valuable.

2. A strong blanket recommendation to any organisation is to log system and user actions: "Make sure you have logs with these N things in them". These logs can be used later for quality assurance by third parties, even when the organisation currently has its attention elsewhere.

3. The DevOps approach to software engineering and operation, with its emphasis on integration of development, quality assurance, and IT operations offers some solutions to bridging the gap between evaluation and commercial relevance. A broad and live contact surface between developers, product team, and quality tests will ensure the feedback enters into the product development process seamlessly. The recommendation from this workshop is that where that process today is vectored towards the handling of development tickets, to insert and accommodate continuous evaluation as a feature, and improved quality metrics as a target notion in that process.

4. Any customer-oriented organisation will have some process for receiving, acknowledging, and acting on customer feedback, requests, worries, and complaints. Some of the items in that feed will be turned into bug reports and development tickets. A strong recommendation is to utilise that pipeline for quality assurance purposes. If evaluation activity feed into that process, its findings cannot be ignored.

5. The process of moving a proof of concept into production involves evaluation of the functions the proof of concept relies on. Those evaluation metrics and measures should be retained when the full scale solution is later implemented, in the form of unit tests or similar continuous evaluation efforts. This means that the formalised process of going from proof of concept to production tool needs to add quality evaluation as a component of development effort.

6. A crucial factor is to bridge social factors within an organisation which may be difficult to discern at the outset. To get product owners and teams to put attention to occasionally disturbing quality evaluation, examples and test items can be based on user data to raise their perceived relevance; user experience driven evaluation efforts from further down the pipeline (which are likely to be accepted throughout the organisation) can be translated into component-level target notions (e.g. coverage of a system is influenced by the recall metrics of a retrieval component); dashboards which display system performance can and should be discretised to provide periodic reports and alerts to raise their priority.

7. A concrete suggestion is to formulate an evaluation clinic at ECIR or SIGIR, inviting a business consultant to give keynote and stakeholders to deliver illustrative examples.

Grateful acknowledgments

This workshop was hosted by BCS-IRSG in Covent Garden, and we are grateful for the help Udo Kruschwitz and John Tait to put things into place. The workshop was partially funded by the ELIAS Grant from the European Science Foundation.