Coopetition in IR Research

Ellen M. Voorhees
National Institute of Standards and Technology
ellen.voorhees@nist.gov

Abstract

Coopetition is defined as competitors cooperating for the common good, an apt description of the community evaluations such as TREC that build infrastructure to support information retrieval (IR) research. This note summarizes the main points of a keynote talk presented at SIGIR 2020 that looks at the benefits and risks coopetition can engender in IR research.

Wikipedia defines coopetition as 'cooperative competition'[1]. The information retrieval (IR) research community has a variety of events variously called community evaluations, challenge problems, shared tasks, and the like that use coopetition to build infrastructure, mostly Cranfield-style test collections, to support its research. One of these events is the Text REtrieval Conference (TREC) project that began almost 30 years ago with the goal of building a single large test collection. Since then, TREC and similar evaluations such as NTCIR, CLEF, and FIRE have built hundreds of collections and have greatly contributed to our understanding of how the Cranfield paradigm succeeds and fails.

This note outlines five benefits and the concomitant risks of using community evaluations to build infrastructure to support research. It is based on a keynote talk I gave at SIGIR 2020 conference, which in turn was based on my experience of managing TREC for more than 20 years. The points will be framed in reference to TREC, because I know TREC best, but generally apply to other evaluations as well. While not universally accepted, in this note (and the talk), I make the fundamental assumption that the Cranfield paradigm is beneficial to the field.

1 Cranfield paradigm

Since the community evaluations build the infrastructure that enables the use of the Cranfield paradigm, I first define what I mean by the Cranfield paradigm and the mechanics of a shared task. The overarching goal is to evaluate the quality of different search techniques to improve search technology by weeding out less effective methods.

The Cranfield paradigm uses a starkly simple definition of a good search result: relevant documents are ranked before non-relevant documents. (This ability is necessary, but not sufficient, for building an effective search system for operational use.) Cranfield tests systems' ranking abilities using test collections. A test collection consists of a document set, a set of queries, and

[1] https://en.wikipedia.org/wiki/Coopetition

relevance judgments that define which documents should be retrieved for which queries. A retrieval system creates a ranked list of documents for each query, where a list is ordered by decreasing likelihood that the document is relevant to the current query. The relevance judgments are used to calculate some effectiveness score over the ranked lists. If retrieval method A creates ranked lists such that the average score over all the queries in the set is better than the average score of the lists created by a second method B, then method A is assumed to be better than method B. Different effectiveness measures provide abstractions of different user tasks.

Test collections are laboratory tools that allow system builders to abstract away many details of a particular search environment to better focus on the aspect of interest. Cranfield is successful because its task abstraction is close enough to real user tasks to be informative, but general enough to be feasible and relatively inexpensive (compared to running a full user study every time a system parameter is tweaked, for example). It purposely does not attempt to model operational necessities in full detail to gain experimental power.

The difficult part of building a test collection is creating the relevance judgments. The judgments cannot be created automatically (if they could, the IR problem would be solved), and useful document set sizes are much too large for a human to look at every document. So some sort of sampling must be used to form the set of documents that a human assessor will judge. This is where the community evaluations come in. Task organizers provide a document set and a set of queries, and then accept submissions of ranked lists of documents from a diverse set of participants. Organizers select top-ranked documents from among the submissions as the set of documents to be judged by the assessor. Participant submissions are then scored based on that set of relevance judgments. Submissions, scores, and judgments sets are finally archived for future use.

2 Benefits and Risks

This section looks at the impact community evaluations have on five aspects of research: state-of-the-art effectiveness, community, methodology, technology transfer, and cost. My claim is that community evaluations are highly beneficial for the research community with regard to these aspects, though there are some risks that need to be mitigated for the full potential to be realized.

State-of-the-art effectiveness: Community evaluations not only document the state of the art on a particular task, they also drive its improvement. For example, retrieval effectiveness approximately doubled in the first six years of TREC as measured by any of the traditional effectiveness measures (e.g., recall, precision, MAP). The improvement comes mostly from the test collections themselves since they allow researchers to experiment at their own pace. The centrality of test collections also poses a risk, however. Any test collection is going to have its own peculiarities, and if a task has only one or two collections, the entire community can overfit to it. The best mitigation strategy for this is to create multiple, distinct collections for a given task. The whole community using the same collection also introduces repeated test issues that likely inflate the apparent statistical significance of retrieval results.

Research community: Infrastructure built through a shared task enables research on that task, which in turn attracts a critical mass of researchers and thus builds a research community.

This infrastructure includes not only a test collection, but also the definition of the evaluation task abstraction. The risk here is that a poor task abstraction wastes the community's time and effort for inconsequential results.

Research methodology: Most community evaluations include a workshop where participants gather to discuss both their findings and the task definition. These meetings are instrumental to defining the way research is done in the area more generally, including appropriate measures and experimental design. A large part of the reason why community evaluations are effective for defining the methodology is their focus on having a broad set of people actually implement the same task. The first running of a TREC track, for example, is generally where one finds out what the *real* problem is.

Technology transfer: One of the main goals of TREC is to compare retrieval methods on a common task because such comparison facilitates consolidation of a wider variety of results than any one research team can tackle alone. Effective approaches are then adapted and incorporated into a variety of search techniques, not only by different research teams, but transfer from research into operational systems as well. The risk associated with this is incentivizing conformity and standardization too early. Multiple collections can help mitigate this risk, as it is a variant of the community overfitting problem.

Infrastructure costs: Building infrastructure does require resources, including both financial resources and researcher time. Centrally-built infrastructure amortizes those costs over the entire community. Avoiding community evaluations to save the cost of running them is a false economy since the lack of infrastructure is much more expensive in the long run. Avoiding participating in a community evaluation to save individual participation costs can be beneficial to those not participating, but the entire ecosystem collapses if too few participate. Participation can be encouraged by streamlining task guidelines and result formats. TREC also rewards participants by giving them earlier access to the data sets (test collections built in TREC X become public in February X+1) and restricting the conference itself to participants only.

3 Conclusion

TREC was an early advocate of coopetition and has demonstrated its utility. While competition can give one a bigger piece of the pie, cooperation makes the whole pie bigger.