QA@INEX 2009: A common task for QA, focused IR and automatic summarization systems

Veronique Moriceau[1], Eric SanJuan[2], and Xavier Tannier[1]

[1] LIMSI-CNRS, University Paris-Sud 11
{moriceau,Xavier.Tannier}@limsi.fr
[2] LIA & IUT STID, Université d'Avignon
eric.sanjuan@univ-avignon.fr

Abstract. QA@INEX aims to evaluate a complex question-answering task. In such a task, the set of questions is composed of factoid, precise questions that expect short answers, as well as more complex questions that can be answered by several sentences or by an aggregation of texts from different documents. Question-answering, XML/passage retrieval and automatic summarization are combined in order to get closer to real information needs.

1 Introduction

The INEX 2009 QA@INEX track aims to compare the performance of QA, XML/passage retrieval and automatic summarization systems on an encyclopedic resource (Wikipedia). The track considers two types of questions: factual questions which require a single precise answer to be found in the corpus if it exists and more complex questions whose answers require the aggregation of several passages. For example, this is the case for questions expecting an answer composed of several items. Current evaluation campaigns artificially restrict list of questions to items present in the same sentence. The reason is that traditional QA systems are not designed to merge answers from different sources, and that human assessment would be made quite harder without this restriction. However, this corresponds to an important user need in several manners:

- Compiling different elements scattered in the collection into a single list of items;
- Finding several valid answers to a single question ("Who is Nicolas Sarkozy?" leads to "French president", "former french interior minister", "Carla Bruni's husband", etc.);
- Gathering different answers with different restrictions: temporal ("Who is the French president?": "Jacques Chirac" from 1995 to 2007, "Nicolas Sarkozy" from 2007), spatial or others.

This is also the case of more complex questions that have not been studied in details so far: see [1], [2, 3] for why questions and [4] for opinion questions).

Questions concerning procedures (in short, “how” questions), reasons (“why”) or opinions can hardly find a complete answer in a single part of a document. For example, concerning opinion questions, a QA system should be able to locate opinions in documents and to produce or generate a synthetic ”answer” in a suitable way.

A extended range of evaluation methods are used to compare QA vs focused IR when a short answers is required (§2) and QA vs summarization systems by extraction on aggregated answers (§3).

2 Short answers

Short parts of text (one or a very few words) are the usual way to answer questions in so-called question-answering systems. Mostly, answers are named entities (person, date, number... answering to factual questions) or short nominal phrases, often representing a definition (*Who was Kurt Cobain? → the leader of Nirvana*; *What is Linux? → an operating system*).

The results are presented as a ranked-list of answers together with an explanation passage or element involving the answer. Therefore participants need to provide:

- A small ordered set (10) of non overlapping XML elements or passages that contains a possible answer to the question.
- For each element or passage, the position of the answer in the passage. They are evaluated by computing their distance to the answer.

This evaluation methodology differs from traditional QA campaigns, where a short answer must be provided besides the supporting passage. This is a major difference in terms of metrics used to rank the participating systems.

In traditional campaigns, an important technical issue for QA system is the boundaries of the short answer in the passage. In the quite simple question *Who is Javier Solana?*, the following passage would be relevant:

Javier Solana, the Secretary General of NATO, has just announced that the bombing of Yugoslavia may start as soon as the next few hours.

A system answering only “the Secretary General” (skipping “of NATO”) as a short answer would be penalized for its incomplete (or *inexact*) answer.

However, this metric does not correspond to a real user need. In a end-user QA application, the obvious way to exhibit the answer is to point directly towards it into the supporting text. In this situation, the user does not need a perfect segmentation of the answer, but rather a good entry point inside the text. He/she is able to estimate the full answer by him/herself, by reading the text surrounding the entry point.

For this reason, we suggest to assess a good answer not through the full/incomplete paradigm, but rather by the distance between the indicated answer entry point and the real one.

This new way to evaluate QA systems has an interesting side effect: it allows focused IR systems to participate in this task using the same evaluation, even if they are unable to extract a short answer or if they have very basic techniques to do so. These systems may simply provide the most relevant and short extracted passages they retrieve, and set an entry point wherever they can in this text. This makes then possible the junction between QA, XML retrieval and other focused IR systems.

3 Long answers

INEX has a thorough experience in evaluating focused retrieval systems, however the QA the "long answer" subtask is new in this context.

Following the first edition of Text Analysis Conference (TAC)[3], that brings together QA and automatic summarization, the idea here is to propose a common task that can be processed by three different kind of systems: QA systems providing list of answers, automatic summarization systems by extraction and focused IR systems.

In this QA task, answers have to be built by aggregation of several passages from different documents on the Wikipedia. The questions themselves can be the same as in the short answer task. Let us consider again the previous example "Who is the leader of Nirvana". The difference with the short answer task is that here we require a short readable abstract of all the information in the Wikipedia related to this question. In this example, the abstract could not only involve references to iconic leader singer of this pop music group, but also on the group itself, on the other members that assumed part of the leadership and that heavily influenced the music style. Passages that explain the terms of the question can also be relevant, by example, why and who decided to take the name of Nirvana for this band.

The maximal length of the abstract being fixed, the systems have to make a selection of the most relevant information. Standard QA systems can produce a list of answers with their support passages. Focused IR systems can return the list of the most relevant XML elements. Note that in this task, IR systems that only retrieve entire documents are strongly handicapped, except if they are combined with automatic summarization systems that builds an abstract of the most relevant documents.

Two main qualities of the resulting abstracts need to be evaluated: readability and informative content.

The readability and coherence is evaluated according to "the last point of interest" in the answer which is the counterpart of the "best entry point" in INEX ad-hoc task. It requires a human evaluation where the assessor indicates where he misses the point of the answers because of highly incoherent grammatical structures, unsolved anaphora, or redundant passages.

The informative content of the answer has to be evaluated according to the way they overlap with relevant passages that will be assessed by participants as in the INEX ad-hoc task. For that we plan to apply recent results on automatic

[3] http://www.nist.gov/tac/publications/2008/

summary evaluation based on the source text. Given a list of relevant passages, these passages can be whole Wikipedia articles, we intend to compare the word distributions in these passages with the word distribution in the long answer following the experiment in [5] done on TAC 2008 automatic summarization evaluation data. This allows to directly evaluate summaries based on a selection of relevant passages without requiring reference summaries written by experts as in TAC. Indeed, such manual summaries based on such large corpus as the Wikipedia would be very difficult to produce. Therefore, this long answer task is a first tentative of evaluating summarization tools on large data.

Given a set R of relevant passages and a text T, let us denote by $p_X(w)$ the probability of finding a word w from the wikipedia in $X \in \{R, T\}$. We use standard Dirichlet smoothing with default $\mu = 2500$ to estimate these probabilities over the whole corpus.

The two metrics of distributional similarity that we implemented in a perl program are the following. The perl program applies these metrics after stemming of the words and relies on an Indri index to estimate a priori probabilities.

- Kullback Leibler divergence:

$$KL(p_T, p_R) = \sum_{w \in R \cup T} p_T(w) \times \log_2 \frac{p_T(w)}{p_R(w)}$$

- Jensen Shannon divergence:

$$JS(p_T, p_R) = \frac{1}{2}(KL(p_T, p_{T \cup R}) + KL(p_R, p_{T \cup R}))$$

Since all answers have to be extracted from the same INEX corpus, we can use smoothing methods that allow to avoid null probabilities. Therefore KL is well founded. JS allows to reduce the impact of smoothing parameters since it is always defined. In [5] this is the metric that obtained the best correlation scores with ROUGE semi automatic evaluations of abstracts used in DUC and TAC. However, since we can compute these probabilities by taking the INEX corpus as referential for the probabilistic space, KL metric should also perform well in this track.

We also implemented the standard cosine distance.

4 Status of the track and time line

We intend to run this track over two years (2009 - 2010). The track is open to new participant teams.

2009 has been devoted to fix the tasks and the overall evaluation methodology based on the corpus, topics and qrels from INEX 2009 ad-hoc track. A first list of questions have been released for test. They all deal with 2009 INEX topics. Hence answers should be part of ad-hoc relevant passages. The process of annotating correct answers among passages is on going by organizers and actual participants.

Based on that we intend to release the software to automatically evaluate content selection for this first set of questions. Results from baseline systems proposed by actual participants will be also released. The track is still open to the submission of baseline runs and each participant is invited to submit at least one. In order to facilitate submissions from Focused IR systems, a perl program that converts a run in INEX ad-hoc submission FOL format into QA format is available.

This will allow to fix in accordance with participants all parameters to be used in metrics to evaluate answers content. In particular, the results of KL and JL metrics for all submitted baseline systems will be available for different smoothing parameters.

In 2010, we shall use the same corpus but participants will be invited to submit a new set of questions on the wikipedia. These questions will not necessarily be related to ad-hoc topics. An additional set of questions on ad-hoc topics will be also proposed by organizers. Once released this 2010 set of questions, participants will have a short time period to submit the results by their systems. This period will be set in accordance with participants. Runs from baseline systems will be also added. Informative content and linguistic quality of answers will be evaluated by participants and organizers based on a short questionnaire.

5 Conclusion

QA@INEX is offering an evaluation framework combining QA, passage retrieval and automatic summarizing by passage extraction. Its main features are the use of the wikipedia as referential, its proximity with INEX ad-hoc task and the introduction of new evaluation metrics.

References

1. Lee, Y.H., Lee, C.W., Sung, C.L., Tzou, M.T., Wang, C.C., Liu, S.H., Shih, C.W., Yang, P.Y., Hsu, W.L.: Complex Question Answering with ASQA at NTCIR 7 ACLIA. In: Proceeding of the 7th NTCIR Workshop Meeting, Tokyo, Japan (dec 2008) 70–76
2. Verberne, S., Boves, L., Oostdijk, N., Coppen, P.A.: Discourse-based answering of why-questions. Traitement Automatique des Langues, Discours et document: traitements automatiques **47**(2) (2007) 21–41
3. Verberne, S., Raaijmakers, S., Theijssen, D., Boves, L.: Learning to Rank Answers to Why-Questions. In: Proceedings of 9th Dutch-Belgian Information Retrieval Workshop (DIR 2009). (2009) 34–41
4. Yu, H., Hatzivassiloglou, V.: Towards Answering Opinion Questions: Separating Facts from Opinions and Identifying the Polarity of Opinion Sentences. In: Proceedings of 2003 Conference on Empirical Methods in Natural Language Processing (EMNLP). (2003) 129–136
5. Louis, A., Nenkova, A.: Performance confidence estimation for automatic summarization. In: EACL, The Association for Computer Linguistics (2009) 541–548