TrainQA: a Training Corpus for Corpus-Based Question Answering Systems

David Tomás, José L. Vicedo, Empar Bisbal, and Lidia Moreno

Abstract—**This paper describes the development of an English corpus of factoid TREC-like question-answer pairs. The corpus obtained consists of more than 70,000 samples, containing each one the following information: a question, its question type, an exact answer to the question, the different contexts levels (sentence, paragraph and document) where the answer occurs inside a document, and a label indicating whether the answer is correct (a positive sample) or not (a negative sample). For instance, TrainQA can be used for training a binary classifier in order to decide if a given answer is correct (positive) to the question formulated or not (negative). To our knowledge, this is the first corpus aimed to train on every stage of a trainable Question Answering system: question classification, information retrieval, answer extraction and answer validation.**

Index Terms—**Question answering, corpus-based systems.**

I. Introduction

EMPIRICIST approach to Natural Language Processing (NLP) suggests that we can learn the complicated and messy structure of language studying large amount of real-life language samples by means of different techniques such as statistical, pattern recognition or machine learning methods. This data-driven approach is based on large corpus, i.e., large body of language data: written texts, spoken discourse, samples of written or spoken language.

Many researchers agree that significant progress can be made in text understanding by attempting to automatically extract information about language from very large corpora. For this reason, many resources have been developed to assist the learning task. These text resources present different levels of annotation that determine the task they are useful for. There are plain corpus like Project Gutenberg[1] that present no extra information, but plain text. There are also corpus like Spanish EFE Press Agency news of 1994 and 1995 (see CLEF[2]), with formatting attributes that identify information about edition, authors, headlines or paragraphs. Finally, there are annotated corpus like Penn Treebank [1] with more elaborated information about part of speech or syntactic structure. All these are general corpora not intended for a concrete task.

In this paper we present a corpus developed to assist data-driven Question Answering (QA) systems. These systems try to obtain exact answers from large corpus to precise questions formulated in natural language. We have developed a corpus of English question-answer pairs suited to train on every stage of a machine learning based QA system: question classification, information retrieval, answer extraction and answer validation.

The corpus consists of more than 70,000 samples. Each of these samples contains information that relates a question with its answer in four different contexts: exact match, sentence, paragraph and document. Every sample is labelled as *positive* or *negative*, depending whether the answer given is correct or not. Negative instances are useful to provide the context in which an extracted answer is incorrect. We obtained a total of 7,598 positive samples and 64,384 negative ones. This way, the corpus can be used to train a binary classifier in order to decide if a given answer is correct (*positive*) to the question formulated or not (*negative*). Moreover, information about question type is also stored to assist the question classification process.

Other corpora have been previously employed to train isolated parts of a QA system. Nevertheless, to our knowledge this is the first corpus that can be used to train all the different components of a QA system and also, the only one that contains positive and negative instances.

This paper is organized as follows: in Section II we introduce the current research related to corpus development and trainable QA systems; Section III describes the samples that make up the corpus; Section IV presents the resources employed to build the corpus and details the generation process; Section V outlines corpus statistics and finally, in Section VI we discuss possible corpus applications and main challenges for future work.

II. Related Work

There are several QA systems that apply machine learning techniques based on corpus of question-answer pairs, covering different stages of the question answering process.

In [2], a corpus of question-answer pairs (called KM database) was developed. Each of the pairs in KM represents a trivia question and its corresponding answer, such as the ones used in the trivia card game. The question-answer pairs were filtered to retain only questions and answers that look

Manuscript received November 23, 2008. Manuscript accepted for publication August 15, 2009.

David Tomás and José L. Vicedo are with the Department of Software and Computing Systems, University of Alicante, Spain (dtomas@dlsi.ua.es, vicedo@dlsi.ua.es)

Empar Bisbal and Lidia Moreno are with Department of Information Systems and Computation, Technical University of Valencia, Spain (ebisbal@dsic.upv.es, lmoreno@dsic.upv.es)

[1] http://www.gutenberg.org

[2] http://www.clef-campaign.org

similar to the ones presented in the TREC task[3]. Finally, 16,228 pairs were obtained in all. Using this corpus as seed, they automatically collected a set of text patterns which are used for answer extraction purposes.

In [3], they built their QA system around a noisy-channel architecture which exploited both a language model for answers and a transformation model for answer/question terms. In order to apply the learning mechanisms, they first built a large training corpus consisting of question-answer pairs of a broad lexical coverage. They collected FAQ pages and obtained a total of roughly 1 million question-answer pairs. They applied this training corpus in the query analysis and answer extraction modules. This system was intended to be applied to non-factoid questions.

The system developed by [4] used a collection of approximately 30,000 question-answer pairs for training, obtained from more than 270 FAQ files on various subjects in the FAQFinder project [5]. They used this corpus to automatically learn phrase features for classifying questions into different types, to generate candidate query transformations, and to evaluate the candidate transforms on target information retrieval systems such as real-world general purpose search engines.

The approach in [6] is heavily inspired by machine learning. Starting from a large collection of answered questions, the algorithms described learn lexical correlations between questions and answers. To serve as a collection of answered questions, they assembled two types of data sets: 1,800 pairs from Usenet FAQs and 5,145 from Call-center dialogues.

All the corpora mentioned above present some of the following problems when employed to train a QA system:

– No question type is given, so that the corpus can not be employed in the question classification stage.
– Negative samples are not provided, which are useful for a classifier to determine the context of incorrect answers.
– The context where the answers occurs is inadequate for different QA stages: too constrained for information retrieval or too loose for answer extraction.

We have developed a corpus that overrides all these problems. It consists of question-answer pairs in XML format that have been obtained from TREC[4] resources (specifically TREC QA track questions and corpora). This way, we have gathered a corpus of factoid TREC-like questions and answers fully oriented to the QA task. Unlike the other approaches, every sample is tagged with a question type that makes them useful for question classification. The corpus presents four different context levels for every answer that make them suitable for every QA stage: document and paragraph context for information retrieval, sentence context for answer validation and exact match for answer extraction. Moreover, our corpus contains correct and incorrect answers, which means that we have pairs labelled as *positive* or *negative* that can be very useful to train binary classifiers. Finally, the number of samples obtained (over 70,000) makes the corpus appropriate for machine learning purpose.

III. Corpus Description

The corpus developed consists of a set of English question-answer pairs samples including the following fields:

– The number of sample, used as identifier.
– The number of question in the TREC set.
– The question itself.
– A question type indicating the class of the question from a taxonomy of fifteen different classes (see [7]) such as LOCATION, PROPER_NAME, EVENT, ORGANIZATION, ACRONYM, . . . This information is useful for the question classification task, where a class or category is assigned to the question proposed.
– The exact answer string. This information can assist the answer extraction process, which allows to obtain nothing but the exact answer to the question formulated.
– The textual context, with the size of a sentence, where the answer was found. This information along with the question, can be employed to train textual entailment systems [8] which can cope with answer validation processes.
– The textual context, with the size of a paragraph, where the answer was found. This information is useful to train a passage retrieval system in order to discriminate between relevant and non relevant paragraphs.
– The identifier of the document where the answer was found. Documents are useful to train good document retrieval or document re-ranking systems to reject non answer bearing documents.
– A label indicating whether the answer is correct (*positive* sample) or incorrect (*negative* sample). This way, binary classifiers can be trained with our corpus in order to determine if an exact answer, a sentence, or a paragraph fit the given question.

Figures 1, 2 and 3 show three different corpus samples for the question "*Who is Tom Cruise married to?*". The question type is PROPER_NAME, indicating that it expects the name of a person as answer. In Fig. 1, the answer given "*Nicole Kidman*" is correct, and the context (sentence and paragraph) justifies it. Consequently, the sample is classified as *positive*.

In the sample included in Fig. 2, "*Nicole Kidman*" is also given as response, but in this case, the context does not support the answer. This sample is therefore classified as *negative*.

In Fig. 3 "*Bill Harford*" is the answer, and besides this is not true, the context where it was extracted from does not justify it anyway. This sample is also classified as *negative*.

IV. Building the Corpus

First subsection describes the resources necessary to build the corpus. The next one describes the process carried out to obtain the set of samples that make up the corpus.

[3] Questions with 10 words or less, and were not multiple choice.

[4] Text REtrieval Conference: http://trec.nist.gov

```
<SAMPLE id="26821" class="POSITIVE">
  <QID>
    1395
  </QID>
  <QUESTION>
    Who is Tom Cruise married to?
  </QUESTION>
  <QTYPE>
    PROPER_NAME
  </QTYPE>
  <ANSWER>
    Nicole Kidman
  </ANSWER>
  <SENTENCE>
    The drama is said to be about a pair of married psychiatrists (played by
    the married Tom Cruise and Nicole Kidman) and their sexual lives, but
    only a few Warner executives, Cruise and Kidman, and Pat Kingsley, a top
    public relations executive, have seen the film.
  </SENTENCE>
  <PARAGRAPH>
    Along the way, Kubrick’s secretive methods generated a continual buzz. Actors
    had to sign agreements not to talk to the press, and shooting scripts were kept
    under strict security. The drama is said to be about a pair of married psychiatrists
    (played by the married Tom Cruise and Nicole Kidman) and their sexual lives, but
    only a few Warner executives, Cruise and Kidman, and Pat Kingsley, a top public
    relations executive, have seen the film.
  </PARAGRAPH>
  <DOCID>
    NYT19990326.0303
  </DOCID>
</SAMPLE>
```

Fig. 1. An example of a *positive* sample from the corpus. Information is separated in different tags: the identifier of the sample (attribute `id` in tag `SAMPLE`), the class indicating whether it is positive or not (attribute `class` in tag `SAMPLE`), the identifier of the question (tag `QID`), the question itself (tag `QUESTION`), the question type (tag `QTYPE`), the exact answer (tag `ANSWER`), the sentence context (tag `SENTENCE`), the paragraph context (tag `PARAGRAPH`) and the document identifier (tag `DOCID`).

```
<SAMPLE id="26824" class="NEGATIVE">
  <QID>
    1395
  </QID>
  <QUESTION>
    Who is Tom Cruise married to?
  </QUESTION>
  <QTYPE>
    PROPER_NAME
  </QTYPE>
  <ANSWER>
    Nicole Kidman
  </ANSWER>
  <SENTENCE>
    The film itself, starring Tom Cruise and Nicole Kidman as a married couple in
    New York on a sexual odyssey, received wildly mixed reviews.
  </SENTENCE>
  <PARAGRAPH>
    The film itself, starring Tom Cruise and Nicole Kidman as a married couple in
    New York on a sexual odyssey, received wildly mixed reviews. After strong box
    office sales in its first weekend, attendance has dropped sharply.
  </PARAGRAPH>
  <DOCID>
    NYT19990326.0303
  </DOCID>
</SAMPLE>
```

Fig. 2. A negative sample. Despite the answer is correct, the context does not justify it.

```
<SAMPLE id="26831" class="NEGATIVE">
    <QID>
      1395
    </QID>
    <QUESTION>
      Who is Tom Cruise married to?
    </QUESTION>
    <QTYPE>
      PROPER_NAME
    </QTYPE>
    <ANSWER>
      Bill Harford
    </ANSWER>
    <SENTENCE>
      The story follows the descent of Bill Harford (Cruise, toothy as ever), a successful young
      doctor on the Upper West Side of Manhattan, into a perilous, secretive netherworld.
    </SENTENCE>
    <PARAGRAPH>
      At the same time ’’Eyes Wide Shut’’ is a sternly anti-erotic movie that regards its sexual
      license with a cold puritanical hauteur. The movie is not a turn-on (it is really a horror
      film without gore), and the sexual chemistry between its married stars, Tom Cruise and
      Ms. Kidman, is tepid at best. The story follows the descent of Bill Harford (Cruise, toothy as
      ever), a successful young doctor on the Upper West Side of Manhattan, into a perilous, secretive
      netherworld. The catalyst is a confession by his wife, Alice (Ms. Kidman), about the fierce,
      unconsummated desire she once felt for a young naval officer. In black-and-white sequences that
      punctuate the movie, Bill torments himself with visions of Alice and her would-be lover in bed
      together, and these images drive him to examine his own wayward impulses.
    </PARAGRAPH>
    <DOCID>
      NYT19990719.0343
    </DOCID>
  </SAMPLE>
```

Fig. 3. A negative sample from the corpus with incorrect answer.

A. The Resources

The resources necessary to build this corpus were obtained from the Question Answering collections in TREC conferences.

In order to collect question-answer pairs, we wanted to focus on questions with exact answers, so only questions formulated from TREC 2002 to TREC 2005 competitions were taken into account. On previous QA tracks (TREC 1999 to TREC 2001), systems were asked for passages instead of exact answers, so we discarded them. For the same reason, only questions in “main” subtask were collected, avoiding “list” or “passage” queries. We finally gathered a collection of 1,505 typical factoid TREC-like questions. The TREC 2004 and 2005 questions sets had to be reviewed as their format slightly differs from the previos competitions. In this case, a *target* was given (i.e. “*Horus*”) and questions referred to that target were formulated (“*What country is he associated with?*”), so that we had to manually reformulate them in other to obtain an homogeneous question set (“*What country is Horus associated with?*”) with no anaphoric references.

We also used the AQUAINT[5] document collection, which is also part of the resources of the TREC QA track. This collection was used to obtain the contexts where answers to the selected TREC questions occurred. It consists of 1,033,461 documents in English with roughly 375 million words, drawn from three sources: the Xinhua News Service (People’s Republic of China), the New York Times News Service, and the Associated Press Worldstream News Service. This document set was used in the last QA tracks, from TREC 2002 to TREC 2005.

Finally, we used the judgement set files from TREC 2002 to TREC 2005. These files contain information about all submissions to the track. A judgement consist of four fields:

- The question number.
- The identifier of the document on the AQUAINT collection that supports the answer.
- The judgement made by the assessors.
- The answer string.

The judgement made by assessors indicates whether the answer is correct, incorrect, inexact or unsupported.“Unsupported” means that the string contains a correct response, but the document returned with that string does not allow one to recognize that it is a correct response. “Inexact” means that the answer string contains a correct answer and the document supports that answer, but the string contains more than just the answer or is missing bits of it. See [9] for detailed description on how answer strings were judged. Figure 4 shows a snippet of these files.

B. The Process

The corpus was semi-automatically obtained from the resources described above, by means of automatic extraction

[5] Linguistic Data Consortium (LDC) catalog number LDC2002T31 and ISBN1-58563-240-6.

TABLE I
CORPUS STATISTICS

Set	Questions	Judgements	Positive	Negative	Total Samples
TREC 2002	500	15,948	1,837	24,818	26,655
TREC 2003	413	9,841	2,359	15,070	17,429
TREC 2004	230	6,235	1,235	8,219	9,454
TREC 2005	362	11,967	2,167	16,277	18,444
TOTAL	**1,505**	**43,991**	**7,598**	**64,384**	**71,982**

```
1395   NYT19991220.0294   -1   Julia Roberts
1395   NYT19991101.0416    1   Nicole Kidman
1395   APW19990712.0006    3   actress Nicole Kidman
1395   NYT19991101.0416    3   actress Nicole Kidman
1395   APW19990712.0006    1   Nicole Kidman
1395   APW19990423.0019    2   Tom Cruise and Nicole Kidman
1395   NYT19990628.0254    2   Nicole Kidman
```

Fig. 4. Judgement file snippet. The third column indicates if the answer is incorrect (-1), correct (1), unsupported (2) or inexact (3).

of the samples and subsequent manual revision of the *positive* ones, as we will describe later.

First, all the factoid questions from TREC 2002 to TREC 2005 QA tracks were collected. These questions were then manually labelled with their respective question type according to the classification presented in [7].

For every question gathered, an automatic process looked into the judgement set files for all its related submissions. These submissions reflect what the systems participating in these QA tracks replied to the questions formulated during the competition. Each judgement was processed following these steps:

1) Read the answer given to the question.
2) Read the judgement made by the assessors.
3) Read the document identifier and try to match the answer in the document.
4) Retrieve and store all the different paragraphs in the document that contain the answer. As every paragraph is already tagged in the AQUAINT corpus, these tags[6] are employed to easily extract them. A sample is generated for every paragraph.
5) Extract from every paragraph the sentence where the answer occurs. The MXTERMINATOR software [10] was employed to detect sentence boundaries.

At this point of the automatic process, we had a set of samples connecting a question with its exact answer and with the different contexts (sentence, paragraph and document) where this answer was found. If the assessors judged the answer as “incorrect”, the sample was labelled as *negative*. If the judgement was “correct”, the sample was labelled as *positive*. In case the judgement was “unsupported”, the sample was labelled as *negative*, since the answer is correct but the context does not justify it.

Judgements labelled as “inexact” demand a special treatment. In this case, the document justifies the answer but this answer does not perfectly fit the user needs as the string contains extra information or is missing bits of it. To solve this problem, the automatic process gathers all the “correct” answers given to the question in the judgement set, and tries to match them in the document where the “inexact” answer was found. For instance, let’s suppose Fig. 4 shows all the judgments for question number “1395”. When the third judgement of the file is processed, the answer given is “*actress Nicole Kidman*”, that was judged as “inexact” for the TREC assessors. In that case, the automatic process looks for all the possible “correct” answers for question “1395” in the judgement set and tries to match them with document “APW19990712.0006”, where the “inexact” answer occurred. In this example, only “*Nicole Kidman*” from the second and fifth judgment is correct, so this exact answer is searched in document “APW19990712.0006”. Samples obtained this way are labelled as *positive* as inexact answers are now substituted with exact ones.

After finishing the automatic process, we had a large set of samples with the information shown in Figs. 1, 2 and 3. But the whole corpus development process is not completed as there was a problem with some pairs that had to be manually reviewed.

There are two different problems with the automatic extraction process. First, in some cases the answer obtained from the judgement set occurred in different paragraphs inside the same document. The automatic process extracted every matching paragraph and created a sample for each one, labelling all of them either as *positive* or *negative* according to the criterion described above. There is no problem if the label assigned is *negative* as we can assure for every sample that the answer is not correct or the paragraph does not support it. The problems arise when the samples are all labelled as *positive*, because we can not guarantee that all the paragraphs matching the answer in the document support it.

Secondly, in some cases there are anaphoric references between paragraphs in the documents so that the answer an its justification appear in different paragraphs. Thus, as we have established, these samples can not be considered *positive* as the context does not justify them.

This ways, all the *positive* samples were set apart for manual

[6] For some reason the paragraphs in the 1998 Associated Press Worldstream News Service corpus are not labelled, thus answers related to this collection were not taken into account.

review in order to decide whether they are correctly labelled as *positive* or must be changed to *negative*. This reviewing task was carried out by two assessors that decided separately if the *positive* label was correctly assigned. A total of 7,598 samples were reassessed with a kappa agreement of 0.94. The expected agreement was computed according to [11], taken as equal for the coders the distribution of proportions over the categories. In case there was no agreement, a third adjudicator made the final determination.

V. Corpus Statistics

The TrainQA corpus has 71,198 samples from 1,505 different questions (47.31 samples per question on average). The number of *positive* samples collected was 7,598, while the number of *negative* samples was 63,384. The amount of *negative* samples (89%) largely exceeds the amount of *positive* ones (11%). We decided to keep this proportion in the corpus since this are the results of real QA systems submissions.

Table I shows the final corpus statistics. For each TREC competition we include the partial results obtained. Last row shows total results. “Set” column indicates which conference provides textual resources. “Questions” indicates the number of questions used to extract question-answer pairs. “Judgements” indicates the number of judgments included in the judgement set files, that is, the total number of submissions made by participants. “Positive” shows the number of samples labelled as *positive*, while “Negative” shows *negative* ones. Finally, column “Total” summarizes the total number of samples gathered, *positive* and *negative*.

The results obtained reflect that the number of samples that we collected from each TREC competition differs. While TREC 2003 and TREC 2005 present similar results (17,429 and 18,444 samples each one), TREC 2002 set largely exceeds TREC 2004 (26,655 and 9,454 respectively). The main factor for this difference is the number of judgements included in the judgment set file. This number of judgments depends on three circumstances:

- The number of questions formulated to the systems. For instance, there are 500 questions in TREC 2002, while there are only 230 in TREC 2004.
- The number of competing systems and the number of runs submitted. In 2002 there were 67 runs, while ’only’ 54 took part in 2003.
- The convergence of the systems: only different judgements are taken into account. If two systems found the same answer in the same paragraph in the same document, only one sample is obtained.

VI. Conclusions and Future Work

Many natural language applications try to automatically extract information from very large corpora in order to learn linguistic phenomena. Corpus-based approaches have demonstrated to ease the adaptation of systems to new languages and domains. In this paper, we have described the development of a corpus intended to assist every stage of corpus-based QA systems. The corpus was semi-automatically obtained, so that the human effort needed to develop it was minimal. We have focused on English resources as they are much more readily available than for other languages. As our data is fully based on TREC QA track resources, the samples obtained perfectly fit the needs of actual QA systems.

A data collection of 71,982 samples was obtained, which seems large enough to train a corpus-based QA system. Every sample relates a question with its question type, its exact answer, and also provides the sentence, paragraph and document context where this answer occurs. Thus, the corpus is suitable to train on every stage of the QA process, where different contexts are required: question types for question classification stage, exact answers for answer extraction stage, sentences for answer validation stage and documents or paragraphs for information retrieval stage.

Another benefit of our approach is that, unlike other similar corpora, we have not only positive samples but also negative ones, providing the context in which an extracted answer is incorrect. For instance, a binary classifier could be trained on this corpus in order to decide whether a possible answer matches the questions formulated or not.

As future work, we will investigate the use of this corpus together with machine learning techniques in order to build versatile and trainable low cost QA systems.

Acknowledgement

This work has been developed in the framework of the project CICYT R2D2 (TIC2003-07158-C04).

References

[1] M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz, “Building a large annotated corpus of english: The penn treebank,” *Computational Linguistics*, vol. 19, no. 2, pp. 313–330, 1994.

[2] D. Ravichandran, A. Ittycheriah, and S. Roukos, “Automatic derivation of surface text patterns for a maximum entropy based question answering system,” in *NAACL ’03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*. Morristown, NJ, USA: Association for Computational Linguistics, 2003, pp. 85–87.

[3] R. Soricut and E. Brill, “Automatic question answering using the web: Beyond the factoid,” *Information Retrieval*, vol. 9, no. 2, pp. 191–206, 2006.

[4] E. Agichtein, S. Lawrence, and L. Gravano, “Learning search engine specific query transformations for question answering,” in *WWW ’01: Proceedings of the 10th international conference on World Wide Web*. New York, NY, USA: ACM, 2001, pp. 169–178.

[5] R. D. Burke, K. J. Hammond, V. A. Kulyukin, S. L. Lytinen, N. Tomuro, and S. Schoenberg, “Question answering from frequently asked question files: Experiences with the faq finder system,” Chicago, IL, USA, Tech. Rep., 1997.

[6] A. Berger, R. Caruana, D. Cohn, D. Freitag, and V. Mittal, “Bridging the lexical chasm: statistical approaches to answer-finding,” in *SIGIR ’00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA: ACM, 2000, pp. 192–199.

[7] E. Bisbal, D. Tomás, L. Moreno, J. L. Vicedo, and A. Suárez, “A multilingual svm-based question classification system,” in *MICAI 2005: Advances in Artificial Intelligence, 4th Mexican International Conference on Artificial Intelligence*, ser. Lecture Notes in Computer Science, A. F. Gelbukh, A. de Albornoz, and H. Terashima-Marín, Eds., vol. 3789. Springer, November 2005, pp. 806–815.

[8] I. Dagan, O. Glickman, and B. Magnini, "Recognizing textual entailment," in *PASCAL Proceedings of the First Challenge Workshop*, Southampton, UK, April 2005, pp. 1–8.

[9] E. M. Voorhees, "The trec-8 question answering track report," in *Eighth Text REtrieval Conference*, ser. NIST Special Publication, vol. 500-246. Gaithersburg, USA: National Institute of Standards and Technology, November 1999, pp. 77–82.

[10] J. C. Reynar and A. Ratnaparkhi, "A maximum entropy approach to identifying sentence boundaries," in *Proceedings of the fifth conference on Applied natural language processing*. Morristown, NJ, USA: Association for Computational Linguistics, 1997, pp. 16–19.

[11] J. L. Fleiss, "Measuring nominal scale agreement among many raters," *Psychological Bulletin*, vol. 76, no. 5, pp. 378–382, 1971.