# Robust Question Answering for Speech Transcripts Using Minimal Syntactic Analysis

Pere R. Comas[1], Jordi Turmo[1], and Mihai Surdeanu[2]

[1] TALP Research Center, Technical University of Catalonia (UPC),
pcomas@lsi.upc.edu, turmo@lsi.upc.edu
[2] Barcelona Media Innovation Center
mihai.surdeanu@barcelonamedia.org

**Abstract.** This paper describes the participation of the Technical University of Catalonia in the CLEF 2007 Question Answering on Speech Transcripts track. For the processing of manual transcripts we have deployed a robust factual Question Answering that uses minimal syntactic information. For the handling of automatic transcripts we combine the QA system with a novel Passage Retrieval and Answer Extraction engine, which is based on a sequence alignment algorithm that searches for "sounds like" sequences in the document collection. We have also enriched the NERC with phonetic features to facilitate the recognition of named entities even when they are incorrectly transcribed.

**Key words:** Question Answering, Spoken Document Retrieval, Phonetic Distance

## 1 Introduction

The CLEF 2007 Question Answering on Speech Transcripts (QAst) track [8] consists of the following four tasks: Question Answering (QA) in manual trascripts of recorded lectures (T1) and their corresponding automatic transcripts (T2), and QA in manual transcripts of recorded meetings (T3) and their corresponding automatic transcripts (T4).

For tasks T1 and T3 we have adapted a QA system and Named Entity Recognizer and Classifier (NERC) that we previously developed for manual speech transcripts [6, 7]. For the handling of automatic transcripts (T2 and T4) we implemented two significant changes: (a) for Passage Retrieval and Answer Extraction we designed a novel keyword matching engine that relies on phonetical similarity (instead of string match) to overcome the errors introduced by the ASR, and (b) we enriched the NERC with phonetic features to facilitate the recognition of named entities (NEs) even when they are incorrectly transcribed.

## 2 Overview of the System Architecture

The architecture of our QA system follows a commonly-used schema which splits the process into three phases performed sequentially: Question Processing (QP), Passage Retrieval (PR), and Answer Extraction (AE).

1M: *"The pattern frequency relevance rate indicates the ratio of relevant documents"*
1A: *"the putt and frequency illustrating the case the ratio of relevant documents"*
2M: *"The host system it is a UNIX Sun workstation"*
2A: *"that of system it is a unique set some workstation"*

**Fig. 1.** Examples of manual (M) and automatic (A) transcripts.

### 2.1 QA System for Manual Transcripts

For the processing of manual transcripts we used an improved version of our system introduced in [6]. We describe it briefly below.

**QP:** The main goal of this component is to detect the type of the expected answer. We currently recognize the 53 open-domain answer types from [4] plus 3 types specific to QAst corpora (i.e., `system/method`, `shape`, and `material`). The answer types are extracted using a multi-class Perceptron classifier and a rich set of lexical, semantic and syntactic features. This classifier obtains an accuracy of 88.5% on the corpus of [4]. Additionally, the QP component extracts and ranks relevant keywords from the question

**PR:** This component retrieves a set of relevant passages from the document collection, given the previously extracted question keywords. The PR algorithm uses a query relaxation procedure that iteratively adjusts the number of keywords used for retrieval and their proximity until the quality of the recovered information is satisfactory (see [6]). In each iteration a Document Retrieval application (Lucene IR engine) fetches the documents relevant for the current query and a subsequent passage construction module builds passages as segments where two consecutive keyword occurrences are separated by at most $t$ words.

**AE:** Identifies the exact answer to the given question within the retrieved passages First, answer candidates are identified as the set of NEs that occur in these passages and have the same type as the answer type detected by QP. Then, these candidates are ranked using a scoring function based on a set of heuristics that measure keyword distance and density[5].

### 2.2 QA System for Automatic Transcripts

The state of the art in ASR technology is far from perfect. For example, the word error rate (WER) of the meetings automatic transcripts is around 38% and the WER of the lectures is over 20%. Figure 1 shows two real examples of common errors when generating automatic transcripts. From the point of view of QA, imperfect transcripts create the following problems: (a) The keywords identified as relevant by QP define the context where the correct answer appears. They are used for PR and AE. When these specific keywords are incorrectly transcribed by the ASR, all these tasks are in jeopardy. (b) Most NEs (candidate answers) appear as proper nouns with low frequency in the corpora. Due to this low frequency it is unlikely that the ASR language models include them. Then it is probable that ASR incorrectly recognizes the NEs relevant for the AE component.

To address these issues specific to automatically-generated transcripts we have developed a novel QA system by changing the PR, AE and NERC components. The main difference between the new PR and AE modules and those used to process manual transcripts is the strategy for keyword searching. Our hypothesis is that an approximated matching between the automatic transcripts and the question keywords, according phinetic similarity can perform better than classical IR techniques for written text. automatic transcripts and question keywords extracted by QP are deterministically transformed to phonetic sequences. Then we use a novel retrieval engine named PHAST, which computes document (or passage or answer context) relevance based on approximated matching of phonetic sequences. PHAST is detailed in Section 4.

## 3 Named Entity Recognition and Classification

As described before, we extract candidate answers from the NEs that occur in the passages retrieved by the PR component. We detail below the strategies used for NERC in both manual and automatic transcripts.

**NERC for Manual Transcripts.** Our initial idea was to use the NERC we developed previously for the processing of speech transcripts [7]. One change from the previous system is that we replaced the existing SVM classifiers with a multi-class Perceptron. To verify the validity of this approach we annotated the NEs that occur in the QAst development corpus with their types (i.e., person, organization, location, language, measure, system/method and time) and used an 80–20% corpus split for training and testing for both lectures and meetings corpora. This experiment indicated that the development data is sufficient for good generalization for meetings (a $F_1$ score of +75 points in the development test partition) but it is insufficient in lectures: 33 points. This is most likely caused by the small size of the development corpus and the large number of topics addressed. To compensate for the insufficient training data we decided to perform a combination of several NERC models for this task. We merged the outputs of: (a) a rule-based NERC developed previously [6], (b) the NERC trained on the existing development data, and (c) the NERC trained on the CoNLL English corpus.[3] We used the above priority ordering for conflict resolution in case of overlapping assignments (e.g., lectures model has higher priority than the CoNLL model). After model combination the NERC $F_1$ score in the development test partition did not improve but the recall did increase, so we decided to use this combination strategy in the testing since recall is paramount for QA

**NERC for Automatic Transcripts.** We used a similar framework for the processing of automatic transcripts: we annotated the development corpora and trained specific NERC models for lectures and meetings. The significant difference is that here we expand the classifiers' feature sets with phonetic attributes. These features are motivated by the fact that even when the ASR incorrectly

[3] `http://cnts.ua.ac.be/conll2002/ner`

transcribes NEs the phonetic structure is by and large maintained in the transcript (e.g. in Figure 1 the name *"Sun"* is recognized as *"some"*). We used an unsupervised hierarchical clustering algorithm that groups tokens based on the similarity of their phonetic sequences. The stop condition of the algorithm is set to reach a local maximum of the Calinski criterion [2]. Then the cluster of each token is added as a feature (e.g. *"Sun"* and *"some"* share the same cluster), which helps the NERC model generalize from the correct to the incorrect transcript. We also added phonetic features that model prefix and suffix similarity.

## 4 The Phonetic Sequence Alignment Algorithm

This section describes PHAST, the phonetic sequence alignment algorithm we used for keyword matching. The same algorithm can be used for PR and identification of answer contexts. PHAST is based on BLAST[1], an algorithm from the field of pattern matching in bioinformatics, which we adapted to work with phone sequences instead of protein sequences. In our case, the input data is a transcript collection $D$ transformed to phonetic sequences and a set of query terms $KW$ also mapped to phonetic sequences.

PHAST is detailed in Algorithm 1. The procedure works as follows: function *detection*() detects subsequences of transcript $d$ at phone number $r$ with moderate resemblance with keyword $w$, then *extension*() computes a similarity score $s$ between $d$ and $w$ at $r$, and *relevant*() judges how this occurrence at $r$ is relevant to term frequency. Function *detection*() uses a deterministic finite automaton (DFA) length $n$ from $w$ while scanning $d$. Given that the ill-transcribed words keep phonetic resemblance with the original words, our hypothesis is that short sequences of $n$ phones will be in the original position. Function *extension*() is a measure of phonetic similarity [3]. We compute the similarity $s$ of two sequences using the edit distance (Levenshtein distance) with a cost function that measures inter-phone similarity. The score $s$ is a bounded non-integer value normalised into the interval $[0, 1]$ Function *relevant*() considers a *hit* any matching with the score above some fixed threshold. In the context of document retrieval,

**Algorithm 1**

**PHAST algorithm**
**Parameter:** $\mathcal{D}$, collection of phonetically transcribed documents
**Parameter:** $\mathcal{KW}$, set of phonetically transcribed keywords

```
1: for all d ∈ D, w ∈ KW do
2:    while h = detection(w, d) do
3:       s = extension(w, h)
4:       if relevant(s, h) then
5:          mark w as matched → update tf(w, d)
6:       end if
7:    end while
8: end for
```

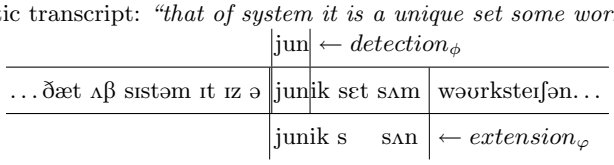


**Fig. 2.** Search of term "UNIX-Sun".

term frequency is computed by adding the scores of these hits. For PR and AE we used all relevant matchings in the algorithms described in Section 2.1. Figure 2 shows an example of how functions *detection* and *extension* are used. The sentence 2A from Figure 1 is transcribed to a sequence of phones. The query word $w$ is the term *"UNIX-Sun"*, which is transcribed as [juniks sʌn].[4] Term $w$ exists in the manual transcript 2M but not in the automatic transcript 2A. In the first step, *detection* finds the 3-gram [jun]. In the second step, *extension* extends it by matching the rest of [juniks sʌn] with the phones surrounding [jun] in the automatic transcript.

## 5 Experimental Results

UPC participated in all four QAst tasks. Initially, each task included 100 test questions, but a few ones were removed due to various problems. The final question distribution was: 98 questions in T1 and T2, 96 in T3, and 93 in T4. In the tasks T1 and T3 we submitted one run using the system described in Section 2.1 ($QA_m$). In the tasks based on automatic transcripts (T2 and T4) we submitted two runs: one using $QA_m$, and another using the system tailored for automatic transcripts as seen in Section 2.2 ($QA_a$) We report two measures: (a) TOP$k$, which assigns to a question a score of 1 only if the system provided a correct answer in the top $k$ returned; and (b) Mean Reciprocal Rank (MRR), which assigns to a question a score of $1/k$, where $k$ is the position of the correct answer, or 0 if no correct answer is found. An answer is considered correct by the human evaluators if it contains the complete answer and nothing more, and it is supported by the corresponding document. If an answer was incomplete or it included more information than necessary or the document did not provide the justification for the answer, the answer was considerated incorrect.

The corpora were pre-processed as follows. We deleted word fragment markers, onomatopoeias, and utterance information in manual transcripts (tasks T1 and T3). Speaker turns in tasks T3 and T4 were substituted by sentence boundaries (this influences our answer ranking heuristics [6]) and the dialog was collapsed into a single document. For T2, all non-word tokens were deleted (e.g., "{breath}"), utterance markers and fragment words were eliminated. Then the documents were pre-processed by a POS tagger, lemmatizer, and NERC.

Table 1 summarizes our overall results. It shows that moving from manual transcripts to automatic transcripts (i.e., the difference of T1/T2, and T3/T4)

[4] We use the international phonetic alphabet (IPA): `www.arts.gla.ac.uk/IPA/`

**Table 1.** Overall results for the four QAst tasks. For task T3 we report scores using a post-deadline submission where some bugs in our output formatting script were fixed.

| Task, System | #Q | MRR | TOP1 | TOP5 | Task, System | #Q | MRR | TOP1 | TOP5 |
|---|---|---|---|---|---|---|---|---|---|
| T1, $QA_m$ | 98 | 0.53 | 50 | 54 | T3, $QA_m$ | 96 | 0.26 | 24 | 27 |
| T2, $QA_a$ | 98 | 0.25 | 24 | 29 | T4, $QA_a$ | 93 | 0.15 | 12 | 17 |
| T2, $QA_m$ | 98 | 0.37 | 35 | 37 | T4, $QA_m$ | 93 | 0.22 | 20 | 22 |

**Table 2.** Distribution of correct answers (TOP5) according to answer type. Org = organization, Per = person, Tim = time, Mea = measure, Met/Sys = method/system, Mat = material, Col = color

| Task and System | Org | Per | Loc | Tim | Mea | Met/Sys | Lan | Sha | Mat | Col |
|---|---|---|---|---|---|---|---|---|---|---|
| T1, $QA_m$ | 10/20 | 8/9 | 4/9 | 7/10 | 12/28 | 10/18 | 3/4 | - | - | - |
| T2, $QA_a$ | 6/20 | 4/9 | 2/9 | 6/10 | 10/28 | 5/18 | 3/4 | - | - | - |
| T2, $QA_m$ | 8/20 | 3/9 | 3/9 | 6/10 | 7/28 | 7/18 | 2/4 | - | - | - |
| T3, $QA_m$ | 5/13 | 8/15 | 6/14 | 1/14 | 4/12 | - | 1/2 | 5/9 | 4/6 | 8/11 |
| T4, $QA_a$ | 2/13 | 3/15 | 2/14 | 1/14 | 2/12 | - | 0/2 | 3/9 | 1/6 | 4/11 |
| T4, $QA_m$ | 3/13 | 2/15 | 3/14 | 1/14 | 4/12 | - | 1/2 | 3/9 | 1/6 | 5/11 |

**Table 3.** Error analysis of the QA system components.

| Task and System | #Questions | QC Correct | PR Correct | QC&PR Correct | TOP1 |
|---|---|---|---|---|---|
| T1, $QA_m$ | 98 | 67 | 82 | 54 | 50 |
| T2, $QA_a$ | 98 | 67 | 80 | 29 | 24 |
| T2, $QA_m$ | 98 | 67 | 76 | 37 | 36 |
| T3, $QA_m$ | 96 | 87 | 73 | 66 | 25 |
| T4, $QA_a$ | 93 | 87 | 52 | 47 | 13 |
| T4, $QA_m$ | 93 | 87 | 58 | 53 | 21 |

yields a drop in TOP1 score of 29% in lectures and 16% in meetings. To our knowledge, this is the first time that such an analysis is performed for QA. It is encouraging to see that our scores are higher than the mean scores observed in TREC 2006 QA evaluation. Surprisingly, the performance drop is smaller for the meetings, even though these transcripts had a higher WER than lectures (38% versus 20%). The explanation is that, because the meetings tasks are harder due to the larger corpus and the more ambiguous question terms, we answer only the "easier" questions in the manual transcripts. Such questions tend to have a larger number of question keywords and answers that appear repeatedly in the collection, so the probability that the system encounter a valid answer even in automatic transcripts is large. In contrast, lecture corpus is very small, so one ASR mistake may be sufficient to lose the only existing correct answer for a given question. Based on these experiments, we can conclude that the QA performance drop follows the WER in small corpora with little redundancy and is smaller than WER in larger corpora with enoug redundancy.

One unexpected result in this evaluation was that the $QA_a$ system performed worse than the $QA_m$ system on automatic transcripts (tasks T3 and T4), even though the $QA_a$ system was designed for automatic transcripts. The explanation

is two fold. First, with our current parameter setting, the PHAST algorithm triggered too many false keyword matches due to a relaxed approximated match. This yielded sets of candidate passages and answers with a lot of noise that was hard to filter out. Second, the NERC training data (i.e., the development corpus) was insufficient to learn correct phonetic generalizations, so many answer candidates were missed in automatic transcripts. Nevertheless, we believe that the architecture of the $QA_a$ system is a good long-term investment because it is the only one of the two systems developed that can address the phenomena specific to automatic transcripts.

Table 2 shows the distribution of correct answers for all tasks according to the answer type. The table indicates that our system had a particularly hard time answering questions in task T3/T4, when the answer type was a NE of types `Org`, `Loc`, `Tim`, or `Mea`. These entity types have a high variation in the corpus and our NERC could not generalize well given the small amount of training data available. This suggests that a better strategy for NERC could be to train an open-domain NERC, where large annotated corpora are available, and use domain transfer techniques to adapt the open-domain system to this domain. The performance drop-off between manual and automatic transcripts is similar in all NE types.

Finally, table 3 summarizes the error analysis of QP, PR, and AE. The "QC Correct" column is the number of questions with the answer type correctly detected by QP. "PR Correct" is the number of questions where at least one passage with the correct answer was retrieved. "QC & PR Correct" is the number of questions where QP prediction is correct *and* PR retrieved a correct passage. We can draw several important observations from this error analysis: QP performs significantly worse for T1 question set than T3 question set. This suggests that in this evaluation T1 questions were more domain specific than T3 questions. Also, PR performs similarly to the state of the art for written text for tasks T1, T2, and T3, but it suffers an important performance hit on task T4, where we processed automatic transcripts with the highest WER (38%). This proves that PR is indeed affected by a high WER. PR using PHAST performed better than the PR with exact keyword match for task T2 and worse for task T4. As previously mentioned, this worse-than-expected behavior of PHAST was due to the many false-positive keyword matches generated in our current setup. We leave the better tuning of PHAST for the various QA tasks as future work. Finally, for tasks T1/T2, when the QA system reaches AE with the correct information (i.e., the "QC & PR Correct" in the table), AE performed very well: we answered most of those questions correctly on the first position. This indicates that both the NERC and the answer ranking performed well. For tasks T3/T4, the story is no longer the same: we suffer the biggest performance hit in AE. We manually inspected these errors and the conclusion was that in most of the cases the fault can be assigned to the NERC, which failed to recognize entity mentions that were correct answers in both manual and automatic transcripts. This problem was mitigated in tasks T1/T2 with a combination of NERC models, which included a rule-based system previously developed for the lectures domain.

## 6 Conclusions

This paper describes UPC's participation in the CLEF 2007 Question Answering on Speech Transcripts track. We were one of the few participants that submitted runs in all four sub-tasks and we obtained the highest overall score. Our best performing runs have TOP1 scores that range from 0.21 (on automatic transcripts with WER of 38%) to 0.51 (on manual transcripts).

In this evaluation we analyzed the behavior of two systems differing in that one is tailored for manual transcripts while the other is tailored for automatic transcripts (uses approximate keyword search based on phonetic distances and a NERC enhanced with phonetic features). In all four tasks we obtained the best performance with the system designed for manual transcripts. This system performed better than expected on automatic transcripts for two reasons: first, it only requires the document collection to be POS tagged, and this technology is robust enough to function well on unperfect automatic transcripts. Second, the query relaxation algorithm adapts well to automatic transcripts: question terms that are incorrectly transcribed are automatically discarded. The system designed for automatic transcripts performed worse than expected because the approximated keyword match algorithm generated too many false-positive, introducing noise in the candidate sets of passages and answers, and also it was impossible for the NERC to detect the correct NEs in the new passages retrieved. Nevertheless, we believe that this approach is a good long-term research direction because it can truly address the phenomena specific to automatic transcripts.

## Acknowledgements

This work has been partially funded by the European Commission (CHIL, IST-2004-506909) and the Spanish Ministry of Science (TEXTMESS project).

## References

1. S. Altschul, W. Gish, W. Miller, E. W. Meyers, and D. J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215:403–410, 1990.
2. T. Calinski and J. Harabasz. A dendrite method for cluster analysis. *Communications in Statistics*, 3, 1974.
3. G. Kondrak. *Algorithms for Language Reconstruction.* PhD thesis, University of Toronto, 2002.
4. X. Li and D. Roth. Learning question classifiers: The role of semantic information. *Journal of Natural Language Engineering*, 2005.
5. M. Paşca. *High-performance, open-domain question answering from large text collections.* PhD thesis, Southern Methodist University, Dallas, TX, 2001.
6. M. Surdeanu, D. Dominguez-Sal, and P. R. Comas. Design and performance analysis of a factoid question answering system for spontaneous speech transcriptions. *Proceedings of the INTERSPEECH*, 2006.
7. M. Surdeanu, J. Turmo, and E. Comelles. Named entity recognition from spontaneous open-domain speech. *Proceedings of the INTERSPEECH*, 2005.
8. J. Turmo, P.R. Comas, C. Ayache, D. Mostefa, S. Rosset, and L. Lamel. Overview of QAST 2007. *Proceedings of the CLEF 2007 Workshop*, 2007.