

# INAOE at QAST 2009: Evaluating the usefulness of a phonetic codification of transcriptions

Alejandro Reyes-Barragán, Luis Villaseñor-Pineda and Manuel Montes-y-Gómez

Laboratory of Language Technologies  
National Institute of Astrophysics, Optics and Electronics (INAOE), Mexico.  
alejandroe@hotmail.com; {villasen, mmontesg}@inaoep.mx

## Abstract

This paper describes the participation of the Laboratory of Language Technologies of INAOE at the QAST 2009 track. This participation was mainly focused on the evaluation of an enriched representation of transcriptions based on their phonetic codification. The idea that motivated the development of this new representation was to reduce the impact of the transcription errors by characterizing words with similar pronunciations through the same phonetic code. The achieved results were unsatisfactory since they indicate the phonetic information had no impact on the answer accuracy, even for those cases considering the transcriptions of lower quality.

## Categories and Subject Descriptors

H.3 [Information Storage and Retrieval]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; H.3.4 Systems and Software; H.3.7 Digital Libraries

## General Terms

Measurement, Performance, Experimentation

## Keywords

Question Answering, Spontaneous Speech Transcriptions, Phonetic Codification.

## 1 Introduction

Question Answering (QA) has become a promising research field whose aim is to provide more natural access to information than traditional document retrieval techniques. In essence, a QA system is a kind of search engine that allows users to pose questions using natural language instead of an artificial query language, and that returns exact answers to the questions instead of a list of entire documents (Burger et al., 2001).

Most QA developments have focused on searching answers for written questions from collections containing written texts. However, due to the great amount of information appearing in spoken documents, such as telephone conversations, meetings, speeches and broadcast news, some recent approaches have begun to address the task of QA in automatic speech transcriptions<sup>1</sup> (Turno et al., 2008).

The initial idea to tackle this problem was to apply the same techniques used in traditional textual-based QA. Nevertheless, this idea did not seem completely appropriate due to existence of (several) mistakes on the automatic transcriptions.

The mistakes generated by an automatic speech recognition (ASR) system may be caused by different circumstances, ranging from the presence of noise in the recording to the presence of unknown terms (e.g., proper nouns that are out of its dictionary), but in general, most of them are approximations of the phonetic production of the original words. Based on this fact, in this paper we propose to enrich the representation of the transcriptions by including their phonetic codification. The purpose of this representation is to reduce the impact of the transcription errors by characterizing words with similar pronunciations through the same phonetic code. In particular, we propose using the Soundex codes (Odell and Russell, 1918) to enrich the representation of transcriptions.

It is important to mention that the idea of using a phonetic codification for indexing automatic transcriptions of speech is not completely new. There was some early attempt to use the Soundex codes for indexing names with the aim of finding all their pronunciation variants (Raghavan and Allan, 2004). Continuing this idea, we applied the phonetic codification indiscriminately to all transcription words, and evaluated the impact of using

---

<sup>1</sup> These transcriptions are the result of applying an automatic speech recognition (ASR) system.

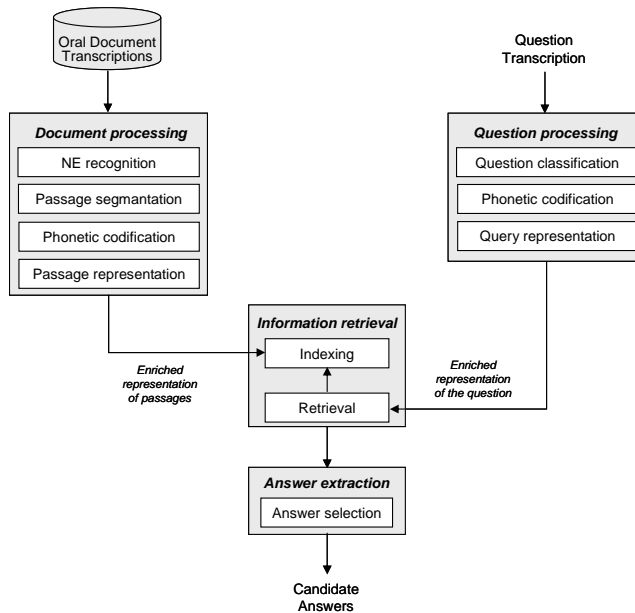


Figure 1. General architecture of the proposed QAST system

this codification in the task of spoken document retrieval (Reyes-Barragán et al., 2008). In this paper we go a step forward and evaluate the usefulness of this codification in the task of QA in speech transcriptions (QAST).

The rest of the paper is organized as follows. Section 2 describes the main modules of the proposed system. Then, Section 3 presents the evaluation results. Finally, Section 4 gives our conclusions and describes some ideas for future work.

## 2 System Description

Figure 1 shows the general architecture of our QAST system. This system follows the traditional architecture, including modules for question processing, passage retrieval and answer extraction; nevertheless, it considers the phonetic codification from queries and passages as additional information.

Given that our participation at QAST 2009 was mainly oriented to evaluate the usefulness of considering the phonetic codification, we intentionally restricted the scope of system to English (tasks T1a and T1b), and to respond questions whose answer is a named entity. In particular, we considered only three types of named entities: dates, quantities and general proper names.

The following sections describe the main modules of the proposed system.

### 2.1 Named Entity Recognition

The first step in the processing of the transcriptions considered the labeling of named entities. This process was achieved by FreeLing (Carreras et al., 2004), a tool that allow recognizing dates, quantities and proper names. Moreover, given that some documents are speeches from the European Parliament, it was necessary to add a set of regular expressions to identify some colloquial expressions related to dates, such as: today, in the future, now, a week ago, in the years to come, etc.

It is important to clarify that we did not apply the same NE recognition process for all given transcriptions<sup>2</sup>. In particular, it was impossible to use FreeLing on C-transcriptions. However, given that our interest was to evaluate the impact of using a phonetic codification, especially from transcriptions of low quality, we decided to apply a naïve method assuming that the NE recognition problem was resolved. This method considered the identification of named entities using an ad-hoc catalogue built from the entities recognized from the automatic transcriptions A and B.

<sup>2</sup> Each task has four different transcriptions, one manual and three automatic (A, B and C), being the transcription A the one with the best quality and the C the one with the lowest.

## 2.2 Passage segmentation

Once finished the process of NE recognition, the transcriptions were segmented into passages of 24 words (named entities were considered as single words), and, after that, adjacent passages were overlapped by 12 words. We considered passages of 24 words because it corresponds to the average length of the sentences of the manual transcriptions.

It is important to comment that the application of this process was motivated by our aim of simplifying the final process of answer extraction.

## 2.3 Phonetic codification

Phonetic codifications attempt to represent words with similar pronunciations by the same code. Among all existing phonetic codification algorithms, Soundex is the most widely known. It was originally proposed for dealing with the problem of having different spelling variations of the same name (e.g., Lewinsky vs. Lewinsky) (Odell and Russell, 1918), and since then it has been applied in several database applications for indexing surnames, for instance, it has been used in the U.S. census.

The Soundex algorithm is based on the phonetic classification of human speech sounds (bilabial, labiodental, dental, alveolar, velar, and glottal), which in turn are based on where we put our lips and tongue to make sounds. The algorithm itself is straightforward since it does not require of backtracking or multiple passes over the input word. This algorithm is as follows:

1. Capitalize all letters in the word and drop all punctuation marks.
2. Retain the first letter of the word.
3. Change all occurrence of the following letters to '0' (zero):  
'A', 'E', 'I', 'O', 'U', 'H', 'W', 'Y'.
4. Change letters from the following sets into the given digit:  
1 = 'B', 'F', 'P', 'V'  
2 = 'C', 'G', 'J', 'K', 'Q', 'S', 'X', 'Z'  
3 = 'D', 'T'  
4 = 'L'  
5 = 'M', 'N'  
6 = 'R'
5. Remove all pairs of equal digits occurring beside each other from the string resulted after step (4).
6. Remove all zeros from the string that results from step (5)
7. Pad the string resulted from step (6) with trailing zeros and return only the first six positions. The output code will be of the form <uppercase letter> <digit> <digit> <digit> <digit> <digit>.

Using this algorithm, both "Robert" and "Rupert" return the same string "R16300", whereas "Rubin" yields "R15000". In our case (refer to Figure 1), we applied the Soundex algorithm to both passages and queries in order to construct their phonetic codifications.

## 2.4 Query and passage representation

Once we obtained the phonetic codification from the passages and the query, we constructed their enriched representations. These representations exclude non-content words but include information about named entities as well as some phonetic codes. Table 1 shows an example of the enriched representation from a passage.

Table 1. Example of an enriched passage representation

Manual transcription (show it for reference)	I am starting to know what <NOM> Frank Sinatra </NOM> must have felt like as his <NOM> farewell </NOM> appearances staggered on into his seventies
Automatic transcription	I am starting to work I do not know what <NOM> Frank Sinatra </NOM> must have felt like as his fellow appearances decided on in the seventies
After elimination of stopwords	starting work <NOM> Frank Sinatra </NOM> felt like fellow appearances decided seventies
Phonetic codification	S36352 W62000 F65200 S53600 F43000 L20000 F40000 A16522 D23300 S15320
Enriched representation	starting work <NOM> Frank Sinatra </NOM> felt like fellow appearances decided <NOM> seventies </NOM> S36352 W62000 F65200 S53600 F43000 L20000 F40000 A16522 D23300 S15320

The enriched representation of a query is very similar to that from a passage; nevertheless, it also includes information about the expected answer type. In our implementation we only consider three basic types of expected answers (date, quantity and proper name), which exactly correspond to the kinds of NEs recognized across the passages. The determination of the expected answer type was carried out in the question classification module (refer to Figure 1) using a set of manually constructed regular expressions.

## 2.5 Passage retrieval and answer extraction

In order to take advantage of all information contained in the enriched representations of passages and queries we decided using Indri (Strohman et al., 2005) as search engine. Indri allowed us to determine a difference importance of each term by assigning them different weights in accordance to their type. For example, for the question “when do we need this strategic program for cohesion?”, whose expected answer type is DATE, we built the following query: (8.0 DATE) (2.0 strategic program cohesion) (1.0 S36322 P62650 C25000), giving much more importance to the expected answer type (a weigh of 8.0), followed by the terms of the query (with a weight of 2.0) and finally to the phonetic codes (a weight of 1.0). These weights were determined empirically from the available training data.

Once retrieved the passages by Indri, we extracted from each passage the entities corresponding to the expected answer type. All extracted answers were ordered in accordance to the position of their source passage (we put first answers from the first passage, then answers extracted from the second passage and so on), and the top five answers were selected as candidate answers.

## 3 Results

This section presents the experiments that were carried out and discusses the achieved results. As was previously mentioned, our system was limited to English tasks T1a and T1b. The difference between these tasks is the kind of queries, while queries from task T1a are written questions, the ones from task T1b correspond to transcriptions from oral spontaneous questions.

In order to evaluate the relevance of the phonetic codifications we sent two runs for each transcription of both tasks: run INAOE1 did not include phonetic information, whereas run INAOE2 employed the proposed enriched representations.

Results corresponding to manual transcriptions are shown in Table 2. For task T1a, that uses written questions, the results are practically the same with and without the phonetic codifications. In other words, the inclusion of the phonetic information was not advantageous, but neither it was prejudicial. In the case of the task T1b, where the questions are oral transcriptions, it is possible to observe an improvement using the phonetic codification, which indicates that the phonetic codification is useful and allow retrieving a greater amount of relevant passages.

Table 2. Results for manual transcriptions

Task	Without phonetic transcription				With phonetic transcription			
	Run id	#Questions with at least 1 correct answer	MRR	ACC	Run id	#Questions with at least 1 correct answer	MRR	ACC
T1a (written questions)	INAOE1	54	0.36	27 %	INAOE2	51	0.36	28 %
T1b (spontaneous oral questions)	INAOE1	35	0.27	22 %	INAOE2	47	0.34	26 %

Tables 3 and 4 show the results of including the phonetic information when automatic transcriptions were used. The results indicate that the phonetic information had no impact on the achieved accuracy, even for those cases considering the transcriptions of lower quality or oral questions<sup>3</sup>. This unexpected behavior may be due to the fact that the phonetic codification reduces the size of the vocabulary, that is, there are less phonetic codes than words. For example, in transcription A there are 4,032 different words and only 2,415 phonetic codes. This reduction increased the number of “irrelevant” passages associated to each query, and, therefore, it produced an unfavorable effect.

<sup>3</sup> It is important to mention that due to the particular treatment of the named entities in transcription C, the results using this transcription are not directly comparable with those from transcriptions A and B. However, it is possible to observe the effect caused by the inclusion of the phonetic codification, which is the main goal of this work.

Table 3. Results for automatic transcriptions from Task T1a

Transcription	Without phonetic transcription				With phonetic transcription			
	Run id	#Questions with at least 1 correct answer	MRR	ACC	Run id	#Questions with at least 1 correct answer	MRR	ACC
A	INAOE1	41	0.3	23 %	INAOE2	42	0.29	22 %
B	INAOE1	29	0.22	17 %	INAOE2	30	0.22	17 %
C	INAOE1	34	0.28	25 %	INAOE2	35	0.28	24 %

Table 3. Results for automatic transcriptions from Task T1b

Transcription	Without phonetic transcription				With phonetic transcription			
	Run id	#Questions with at least 1 correct answer	MRR	ACC	Run id	#Questions with at least 1 correct answer	MRR	ACC
A	INAOE1	40	0.3	24 %	INAOE2	41	0.29	23 %
B	INAOE1	30	0.22	16 %	INAOE2	31	0.22	16 %
C	INAOE1	33	0.28	25 %	INAOE2	34	0.27	23 %

## 4 Conclusions

The present work explored the impact of the phonetic codification in the task of question answering in speech transcriptions. The proposed system enriched the representation of the automatic transcriptions with their corresponding phonetic codes (Soundex codes). To focus our participation in the evaluation of the phonetic codification, it was limited to questions whose answer was a named entity, simplifying in this way the processes of passage retrieval and answer extraction.

As it can be noticed from the results section, the phonetic codification –at least in the way we implemented– did not produce any improvement on the answer accuracy. Despite these results, the use of the phonetic codification should not be discarded completely; it is necessary to make a deeper analysis in order to identify its possible advantages. This is supported by the fact that we achieved some improvement when the phonetic codification was applied to manual transcriptions from the task T1b. In this case the performance was almost the same to that achieved with written questions, indicating that it was possible to decrease –in some degree– the error introduced by the automatic speech recognition system.

**Acknowledgements.** This work was done under partial support of CONACYT (Project Grant 83459). We also like to thank to the CLEF organizing committee for the resources provided.

## References

1. Burger, J. et al. Issues, Tasks and Program Structures to Roadmap Research in Question & Answering (Q&A), in TREC 10, NIST 2001.
2. Carreras, X., Chao, I., Padró, LL., Padró, M. FreeLing: An Open-Source Suite of Language Analyzers. Proceedings of the 4th International Conference on Languages Resources and Evaluation (LREC 2004). European Language Resources Association, 2004.
3. Odell, M.K., Russell, R.C.: U.S. Patent Numbers 1261167 (1918) and 1435663 (1922). Washington, D.C.: U.S. Patent Office, 1918.
4. Raghavan, H., Allan, J. Using Soundex Codes for Indexing Names in ASR documents. In Proceedings of the Workshop on Interdisciplinary Approaches to Speech Indexing and Retrieval at Human Language Technology Conference and North American chapter of Association of Computational Linguistics, Boston, MA, USA, pp. 22–27, 2004.
5. Reyes-Barragán, M. A., Villaseñor-Pineda, L., and Montes-y-Gómez, M. A Soundex-based Approach for Spoken Document Retrieval. Mexican International Conference of Artificial Intelligence, MICAI 2008. Lecture Notes in Artificial Intelligence 5317, Springer 2008.
6. Strohman, T., Metzler, D., Turtle, H., Croft, W.B. Indri: A Language-Model based Search Engine for Complex Queries. In Proceedings of the International Conference on Intelligence Analysis, McLean, VA, May 2005.
7. Turmo, J., Comas, P.R., Rosset, S., Lamel, L., Moureau, N., and Mostefa, D. Overview of QAST 2008. Proceedings of the CLEF 2008 Workshop on Cross-Language Information Retrieval and Evaluation, 2008.