Lexical-phonetic automata for spoken utterance indexing and retrieval

Julien Fayolle[1], *Murat Saraçlar*[2], *Fabienne Moreau*[1], *Christian Raymond*[1] *and Guillaume Gravier*[1]

[1]IRISA (INRIA , University of Rennes 2, INSA, CNRS), Rennes, France
[2]Department of Electrical and Electronic Engineering, Boğaziçi University, Istanbul, Turkey
firstname.lastname@irisa.fr, murat.saraclar@boun.edu.tr

Abstract

This paper[1] presents a method for indexing spoken utterances which combines lexical and phonetic hypotheses in a hybrid index built from automata. The retrieval is realised by a lexical-phonetic and semi-imperfect matching whose aim is to improve the recall. A feature vector, containing edit distance scores and a confidence measure, weights each transition to help the filtering of the candidate utterance list for a more precise search. Experiment results show that the lexical and phonetic representations are complementary and we compare the hybrid search with the state-of-the-art cascaded search to retrieve named entity queries.

Index Terms: information retrieval, speech indexing, lexical-phonetic automata, confidence measures, edit distances, supervised learning

1. Introduction

Spoken content retrieval [1] relies on the fields of automatic speech recognition (ASR) and information retrieval (IR). However, IR tools made for text are not adapted to automatic transcripts which are particularly incomplete and uncertain. Even if in-vocabulary words (IV) are usually well-recognized, these transcripts contain many recognition errors affecting notably out-of-vocabulary words (OOV) and named entities (NE) that convey important discourse information (*e.g.*, person names, localisations, organisations) necessary for IR. Two kinds of approaches can be used to attenuate these drawbacks by either improving the recall or the precision. First, the recall can be improved by using a lower level of representation consisting in sub-words (*e.g.*, syllables, phonemes) to represent OOV words and, more generally, all types of lexical errors. Representations denser than a simple transcript can also be used, such as graphs, confusion networks and N-best lists. Second, the precision can be improved by filtering out noisy parts of the recognition thanks to meaningful features (*e.g.*, confidence measures). We are interested in combining the two approaches for a task of spoken utterance retrieval.

[1]This work was partly achieved as part of the Quaero Programme, funded by OSEO, French State agency for innovation.

Spoken utterance retrieval consists in retrieving, in a spoken content set, all the segments (called utterances) containing a given textual query. Two strategies are used in state-of-the-art systems to combine efficiently both lexical and phonetic levels for searching. The first one considers two separated indexes used in "cascade", *i.e.*, the search is, by default, based on the lexical index and can fall back on the phonetic one if necessary [2]. This limits the usage of the phonetic index, rather noisy, only to mis-recognized queries. The second approach models the two levels in one hybrid index [3, 4], offering the advantage of a hybrid matching between the query and the index.

The proposed method takes up the idea of a hybrid index because it can tolerate lexical-phonetic matchings that are impossible with two separate indexes. The index structure is based on automata as they can represent all types of ASR outputs. The originality of the method consists in the weighting of automaton transitions with a vector of different features that can be used to estimate the relevance of the candidate utterances for a given query. The features used include : edit distance scores (counts of correct symbols, deletions, insertions, substitutions) indicating the imperfection of the matching between the query and the index; and a lexical-phonetic confidence measure indicating the reliability of the recognized symbols. The experiments conducted compare the performances between the cascaded and hybrid searches to retrieve named entity queries. We present first the proposed method (section 2), then the results of the experiments (section 3) and finally conclude the paper (section 4).

2. Method

The proposed method is based on the general indexing of weighted automata presented by Allauzen *et al.*[5] and adapted for the case of lexical-phonetic automata (see figure 1 for an overview of the method). From the ASR outputs, we build the lexical-phonetic automata to be indexed (section 2.1). The textual query is phonetized and converted into a lexical-phonetic automaton as well. A more or less imperfect matching is possible by composing successively the query, an edit transducer and the in-

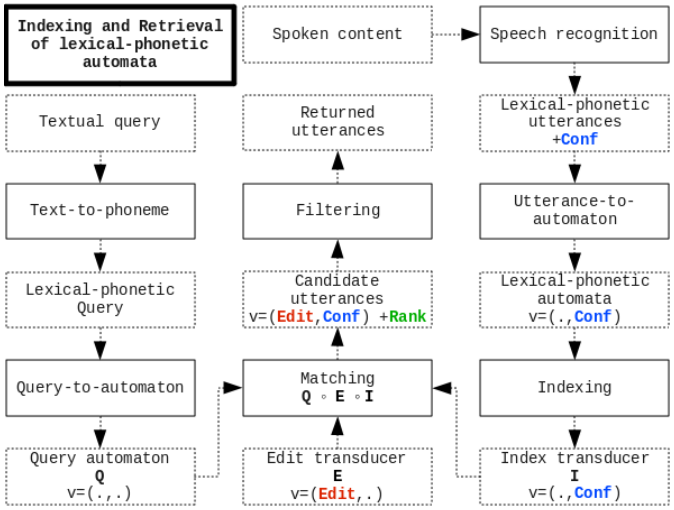

Figure 1: Overview of the proposed method.

dex (section 2.2). This process returns a list of candidate utterances that can be filtered thanks to the feature vector weighting each utterance (section 2.3).

2.1. Lexical-phonetic automata

In this paper, a lexical-phonetic automaton simply denotes a weighted finite-state automaton whose symbols are either from a lexical alphabet Σ^{lex} or a phonetic alphabet Σ^{ph}, and whose weights are multi-dimensional. Thus, a lexical-phonetic automaton can have concurrent lexical and phonetic paths weighted by a vector of various features (*e.g.*, see figure 2). If defined over the tropical semi-ring, then the weight of a path is the sum of its transition weights and the shortest path is the one with the minimum weight. This minimum weight can always be found only if the weights are always comparable, *i.e.*, if they are totally ordered. This is precisely the case when the lexicographic order (also known as the alphabetical order) is considered as in [6]. Each transition corresponds to a symbol (either lexical or phonetic) recognized between the start time t_s and the end time t_e with an associated confidence measure c. The weight of the transition is the following :

$$v = (0, 0, 0, 0, 0, w_{conf}^{lex+ph} = -(t_e - t_s).log(c))$$

where w_{conf}^{lex+ph} is the lexical-phonetic confidence score because it is common to both lexical and phonetic levels. The confidence score is proportional to the duration of the symbol so that concurrent lexical-phonetic paths of different numbers of symbols can be comparable.

Once built, the automaton is turned into a corresponding factor transducer that accepts all the sub-sequences of the automaton in input and gives the utterance identifier in output. The index consists in the union of all the factor transducers (as presented in [5]).

2.2. Lexical-phonetic matching

The matching between the query Q and the index I can be realised by the simple automaton-transducer composition $Q \circ I$. It is however possible to get a more flexible matching using an edit transducer E by the successive composition $Q \circ E \circ I$ [7]. We present three types of lexical-phonetic edit transducers corresponding to perfect, imperfect and semi-imperfect matchings. Their aim is to compute the edit distance scores in the vector

$$v = (w_{cor}^{lex}, w_{cor}^{ph}, w_{del}^{ph}, w_{ins}^{ph}, w_{sub}^{ph}, 0)$$

consisting in the counts of correct words, correct phonemes, and phonetic deletions, insertions and substitutions.

The perfect matching transducer only counts correct words and phonemes. The count of correct words is chosen to be the first dimension of the vector in order to favour the lexical matching rather than the phonetic matching when both are possible. No imperfections are allowed, which makes this transducer particularly restrictive.

The imperfect matching transducer is able to count, besides correct words and phonemes, also phonetic deletions, insertions and substitutions. Its problem is that the matching is done without any constraints and, thus, all imperfections are tolerated (even paths with no correct symbols), which makes this transducer quite greedy.

A good trade-off between the two previous extreme approaches can be to count the imperfections under certain constraints. The proposed semi-imperfect matching transducer takes into account the a priori phonetic variability to limit the imperfection possibilities : "in a sliding window of α phonemes, the rate of correct phonemes must be greater than ρ". In this paper, the parameters are arbitrarily set to $\alpha = 2$ and $\rho = 1/2$ for preliminary experiments. Figure 3 illustrates these three types of transducers for a small lexical-phonetic alphabet.

2.3. Filtering of candidate utterances

After matching and projection on the output label, we obtain a list of weighted utterances ranked according to the lexicographic order. Thus, each candidate utterance is associated to a vector of 7 features :

$$f = (rank, w_{cor}^{lex}, w_{cor}^{ph}, w_{del}^{ph}, w_{ins}^{ph}, w_{sub}^{ph}, w_{conf}^{lex+ph})$$

Determining if an utterance contains (or not) the query from these features can be posed as a binary classification problem solvable by any learning method (*e.g.*, decision trees). Then, the estimated probability of an utterance to contain the query is turned into a binary decision with a threshold set according to the desired recall-precision trade-off.

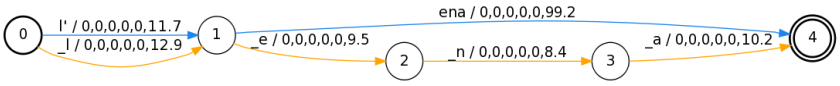

Figure 2: Example of a lexical-phonetic automaton : accepting the lexical path "l' ena", the phonetic path "_l _E _n _a", and the lexical-phonetic paths "l' _E _n _a" and "_l ena"; and weighted by a vector of 6 different features.

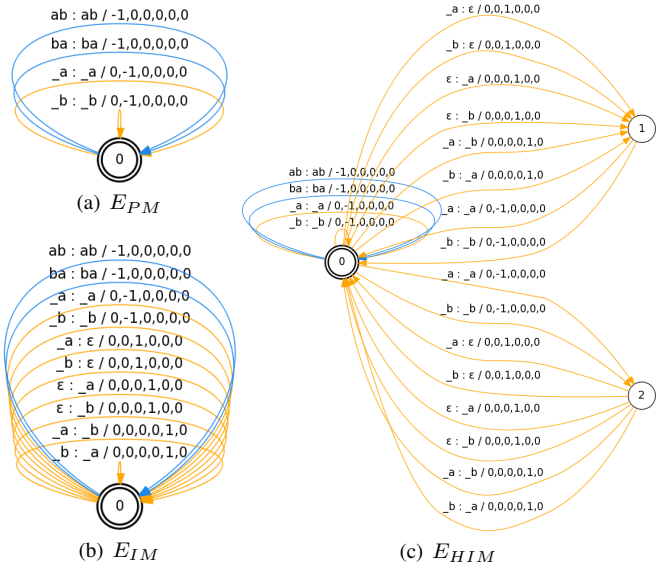

Figure 3: Edit transducers for a lexical-phonetic matching that is perfect (a), imperfect (b) or semi-imperfect (c) where $\Sigma^{lex} = \{ab, ba\}$ and $\Sigma^{ph} = \{\_a, \_b\}$.

3. Experiments

In this section, we present the necessary experimental setup (section 3.1) to implement the proposed method and carry out two experiments, one on the complementarity of the lexical and phonetic levels (section 3.2) and a second one on spoken utterance retrieval (section 3.3).

3.1. Setup

The audio data used for the experiments consists of 6 hours of French radio broadcast news material extracted from the ESTER2 corpus [8] containing reference transcripts with manually annotated named entities. The ASR system is a large vocabulary (65k words) transcription system for which the word error rates on this corpus vary between 16.0% and 42.2%. The data are automatically segmented into 3447 utterances. The N-best hypotheses are then re-scored using a morpho-syntactic tagger [9]. The lexical level is made only of the 1-best hypothesis. The phonetic level is obtained by forced alignment between the audio signal and the pronunciation of the lexical level. Lexical and phonetic confidence measures are calculated from the a posteriori probabilities and the entropy between the different hypotheses [10]. The automata are implemented based on OpenFST[2] and the size of the lexical, phonetic and hybrid indexes are 9.9, 32.8 and 47.6 MB respectively. To avoid matching problems that might appear due to morphological variations, words are turned into lemmas with TreeTagger[3]. To estimate the probability of an utterance to contain the query, we used a bagging over 20 decision trees (Bonzaiboost[4]). The evaluation is done according to a 5-fold cross-validation using 80% of the candidate set for training and 20% for testing. The queries are all named entities extracted from the transcripts of reference. The pronunciation of the query is given by the phonetic lexicon ILPho[5]. If a certain word doesn't belong to the lexicon, multiple pronunciations of it are generated by the phonetizer Lia_phon[6]. In addition to the usual sets of IV and OOV queries, we propose a third set of queries made of both IV and OOV words (*e.g.*, an IV first name followed by an OOV family name). These mixed IV/OOV queries are interesting because they represent an intermediate level of difficulty (a priori more difficult than IV queries but less difficult than OOV ones) and they are more frequent than the OOV queries. Table 1 shows the query distribution. To evaluate the performance of spoken utterance retrieval, we use the mean average precision (MAP) and the precision at N (P@N) where N is the number of the expected relevant utterances for a given query.

3.2. Complementarity of lexical and phonetic levels

This preliminary experiment consists in measuring the quality of the lexical and phonetic representations and their complementarity. For each utterance, we align the lexical-phonetic automata of reference and hypothesis with an imperfect edit transducer to obtain Table 2, which gives the correct symbol rate on named entities. On the one hand, the lexical level is used on areas correctly recognized. On the other hand, the phonetic level is only used on mis-recognized areas. We note that 73.89% of the lemmas are well recognized. For the mis-recognized lemmas, we can fortunately fall back on the phonetic level for which 67.73% of the phonemes are correct. This justifies the combination of lexical and phonetic levels to search for named entities.

[2]http://www.openfst.org/
[3]http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/
[4]http://bonzaiboost.gforge.inria.fr/
[5]http://catalog.elra.info/product_info.php?products_id=760
[6]http://www.atala.org/LIA-PHON

#words	1	2	3	4	5	6	7+	total
IV	209	276	125	73	29	24	34	770 (68%)
OOV	76	43	1	.	.	.	.	120 (10%)
IV/OOV	.	120	73	29	11	8	6	247 (22%)

Table 1: Query distribution in function of the type and the length in number of words.

NE terms	% lemmas in reference	%correct lemmas	%correct phonemes in erroneous areas
IV	93.57	78.97	67.34
OOV	6.43	0.00	68.54
Overall	100.00	73.89	67.73

Table 2: Complementarity of lexical and phonetic representations for named entities.

Evaluation		MAP								P@N							
Matching		Perfect				Semi-Imperfect				Perfect				Semi-Imperfect			
Index		lex	ph	cas	hyb	lex	ph	cas	hyb	lex	ph	cas	hyb	lex	ph	cas	hyb
IV	**th-conf**	.634	.577	.673	.577	.634	.015	.047	.013	63.2	64.3	65.5	64.1	63.3	29.3	*67.1*	27.6
	dt-all	.631	.646	*.677*	*.681*	.629	*.693*	*.713*	***.729***	63.6	64.9	*65.9*	*65.9*	63.2	*74.5*	*73.7*	***74.8***
OOV	**th-conf**	.000	.036	.036	.036	.000	.001	.001	.001	6.6	12.6	12.6	12.6	6.6	8.1	8.1	8.1
	dt-all	.000	*.053*	*.053*	*.053*	.000	***.139***	***.139***	***.139***	6.6	12.0	12.0	12.0	6.6	***27.2***	***27.2***	***27.2***
IV/OOV	**th-conf**	.000	.024	.024	*.029*	.000	.001	.001	.001	16.5	19.5	19.5	19.5	16.5	*25.0*	*25.0*	*24.5*
	dt-all	.000	*.024*	*.024*	*.024*	.000	***.256***	***.256***	*.250*	16.5	19.5	19.5	19.5	16.5	***41.2***	***41.2***	*40.2*
OVERALL	**th-conf**	.523	.479	.556	.478	.523	.009	.015	.008	47.1	49.1	49.8	48.9	47.1	26.3	*52.0*	25.4
	dt-all	.520	.540	*.568*	*.570*	.519	*.610*	*.637*	***.650***	47.4	49.2	*50.0*	*50.1*	47.3	*62.8*	*62.6*	***63.5***

Table 3: Spoken utterance retrieval results : baseline, *better than the baseline*, **best result(s)**.

3.3. Spoken utterance retrieval

The goal of this experiment is to compare the spoken utterance retrieval for different settings. We perform the search using either a lexical index, a phonetic index, both indexes in cascade or a hybrid index. The queries are IV, OOV or IV/OOV while the matching is perfect or semi-imperfect. The imperfect matching has been discarded because it is too greedy. Two filtering methods are considered using a simple threshold either over the lexical-phonetic confidence score (th-conf) or over the probability estimated by the decision trees using all the features (dt-all). The baseline corresponds to the cascade search using a perfect matching and a th-conf filtering. Table 3 reports the obtained performances.

Generally, we first notice that the baseline can easily be improved for all types of queries using a semi-imperfect matching with the dt-all filtering (the th-conf filtering is not sufficient). Second, the hybrid search using the dt-all filtering always performs better or equally than both lexical and phonetic searches. This proves that the hybrid combination is justified.

More specifically, the hybrid search obtains the best results for IV queries. For OOV queries, the hybrid, cascaded and phonetic search are equivalent as they can only use the phonetic level. For mixed IV/OOV queries, it is surprising that the phonetic and cascaded searches are better than the hybrid one. This is due to the fact that the ranking gives too much importance to lexical match even if this one is not really relevant (mis-recognized or very frequent words). We think that adding a tf*idf score in the feature vector will help to deal with these cases.

Finally, the hybrid search (with the semi-imperfect matching and the dt-all filtering) offers the best overall performances.

4. Conclusion

We have presented a method to index lexical-phonetic automata for spoken utterance retrieval. The results demonstrates the complementarity of the lexical and phonetic levels (extracted from the 1-best speech recognition hypothesis) and the advantage of using a hybrid index, a semi-imperfect matching and a supervised filtering (combining edit distance scores and a confidence measure).

5. References

[1] C. Chelba, T. J. Hazen, and M. Saraclar, "Retrieval and browsing of spoken content," *Signal Processing Magazine, IEEE*, vol. 25, no. 3, pp. 39–49, 2008.

[2] M. Saraclar and R. Sproat, "Lattice-based search for spoken utterance retrieval," in *HLT-NAACL'04*, 2004, pp. 129–136.

[3] T. Hori, I. L. Hetherington, T. J. Hazen, and J. R. Glass, "Open-vocabulary spoken utterance retrieval using confusion neworks," in *ICASSP'07*, 2007, pp. 73–76.

[4] P. Yu and F. Seide, "A hybrid-word/phoneme-based approach for improved vocabulary-independent search in spontaneous speech," in *Interspeech'04, Korea*, 2004, pp. 293–296.

[5] C. Allauzen, M. Mohri, and M. Saraclar, "General indexation of weighted automata - application to spoken utterance retrieval," in *HLT/NAACL'04*, 2004, pp. 33–40.

[6] D. Can and M. Saraclar, "Lattice indexing for spoken term detection," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 19, no. 8, pp. 2338–2347, 2011.

[7] M. Mohri, "Edit-distance of weighted automata," in *CIAA'02*. Springer Verlag, 2002, pp. 1–23.

[8] S. Galliano, G. Gravier, and L. Chaubard, "The ESTER 2 evaluation campaign for the rich transcription of French radio broadcasts," in *Interspeech'09*, 2009, pp. 2583–2586.

[9] S. Huet, G. Gravier, and P. Sébillot, "Morpho-syntactic post-processing of n-best lists for improved french automatic speech recognition," *Computer Speech and Language*, no. 24, pp. 663–684, 2010.

[10] T.-H. Chen, B. Chen, and H.-M. Wang, "On using entropy information to improve posterior probability-based confidence measures," in *ISCSLP'06*, 2006, pp. 454–463.