Comparison of Different Phone-based Spoken Document Retrieval Methods with Text and Spoken Queries

Nicolas Moreau, Shan Jin, Thomas Sikora

Department of Communication Systems
Technical University of Berlin, Germany
[moreau, shan, sikora]@nue.tu-berlin.de

Abstract

This study compares four phone-based spoken document retrieval (SDR) approaches. In all cases, the indexing and retrieval system uses phonetic information only. The first retrieval method is based on the vector space model, using phone 3-grams as indexing terms. This approach is compared with 2 string-matching methods. A fourth method, combining the VSM approach with the slot detection step of string-matching techniques is proposed. This method is tested on a collection of short German spoken documents, using three different sets of queries: text queries, clean spoken queries and noisy spoken queries.

1. Introduction

Audio streams of multimedia documents often contain spoken parts that enclose a lot of semantic information. This information, called *spoken content*, consists of the actual words spoken in the speech segments of an audio stream. As speech represents the primary means of human communication, a significant amount of the useable information enclosed in audio-visual documents may reside in the spoken content. In the past decade, the spoken content extraction by means of automatic speech recognition (ASR) systems has therefore become a key challenge for the development of efficient methods to index and retrieve audio-visual documents.

This study presents an ASR-based system for the indexing and retrieval of German spoken documents. Our indexing system extracts phonetic information from speech through a phone recognizer. The resulting transcriptions contain a lot of recognition errors. In this context, specific retrieval methods are required.

In a previous work [1], we have proposed a spoken document retrieval (SDR) approach that exploits the phone confusion statistics in order to expand the phonetic representation of the documents. This method was based on the well-known vector space model (VSM) and used phone 3-grams as basic indexing units.

This study compares this approach with 2 string-matching methods described in [2]. A fourth method, combining a slot detection technique with the VSM approach is proposed. Moreover, we do not only use text queries as in [1]. The case of spoken queries is also investigated.

The paper is structured as follows. Section 2 describes the indexing system. The benefits and drawbacks of indexing spoken documents with phones are discussed. In section 3, we present the different retrieval methods that were tested. Finally, experimental results are reported in section 4.

2. Spoken Document Indexing

In this paper, indexing is the process of generating spoken content descriptions of spoken documents. This study focuses on the use of phonetic indexing terms.

2.1. Phonetic indexing

Traditional SDR systems consist in linking a LVCSR system (large vocabulary continuous speech recognition) with a traditional text retrieval system [3]. But word-based retrieval approaches face the problem of having to know *a priori* the vocabulary to search for. A very large recognition vocabulary is necessary to cover the growing and diverse message collections. Furthermore, the derivation of complex language models, requiring huge amounts of training data, is necessary for reasonable quality LVCSR systems.

The use of sub-word indexing terms is a way of avoiding these difficulties [2][4][5]. First, it dramatically restrains the set of indexing terms needed to cover the language. Furthermore, it makes the indexing and retrieval process independent from any word vocabulary, virtually allowing for the retrieval of any query term (*open-vocabulary* retrieval).

In this study, we will only use phonetic indexing units. The indexing system we developed does not require any *a priori* word vocabulary. However, sub-word indexing approaches have a major drawback. They have to cope with high error rates, much higher than the word error rates of state-of-the-art LVCSR systems. The error rate of a phone recognition system, for instance, is typically between 30% and 40%. The approaches presented in this paper try to compensate for the indexing inaccuracy by exploiting the probabilities of phone recognition errors.

Besides, the use of only phonetic indexing might lose discrimination power between relevant and irrelevant documents when compared to word indexing, because of the exclusion of lexical knowledge. However this study focuses on a simple SDR task with short documents and single word queries. In that case, we think that the use of a simple, vocabulary-independent phone recogniser is a reasonable indexing approach.

2.2. Phone recognizer

The language considered in this study is German. We used a set of 42 phones modeled by context independent HMMs having between 2 and 4 states (depending on the phone). The observation functions are multi-gaussians with 128 modes per state and diagonal covariance matrices. The 39-dimensional observation vectors consist of 12 mel-frequency cepstral coefficients (MFCCs), the energy, and the first and second derivatives. The HTK toolkit was used for training the HMMs

on the German "Verbmobil I" (VM I) corpus, a large collection of spontaneous speech from many different speakers.

Phone recognition is performed without any lexical constraints. The phone HMMs are looped according to a bigram language model (LM), which was trained from the transcriptions of the "Vermobil II" corpus. The set of indexing symbols was reduced by merging some acoustically similar phone classes. We mapped our 42 phones to 32 German "phonemes" [1]. For more convenience, we will continue to use the term "phone" instead of "phoneme".

The recogniser has been tested on the 14th volume of the German VM I corpus (VM14.1) which had not been used for training. We obtained a phone error rate (PER) of 43.0%. This test also provided a set of phone recognition error probabilities that were stored together with the phonetic representations of the documents.

3. Spoken Document Retrieval

Once the spoken document database has been indexed through the phone recognizer, it can be processed by a SDR system.

3.1. Queries

In this study, we will only consider the case of single-word queries. Two types of query were experimented:

- *Text query*. The query word is transformed into a sequence of phonetic units so that it can be matched against the phonetic representations of the documents. Words are generally transcribed by means of a pronunciation dictionary.
- *Spoken query*. A phone recognizer is used to generate a sub-word transcription. This makes the system totally independent from any pronunciation vocabulary, but recognition errors are introduced in the query too.

Once a query word has been transcribed into a phone sequence Q, it is matched against each phonetic document representations D by means of the techniques described in the following sections. A relevance score $S(Q,D)$ is generated, reflecting how *relevant* is D with respect to Q (i.e. how likely will D satisfy the user's request). The relevance scores are finally used for ranking the documents, in order to output the most relevant ones.

3.2. Vector space model using 3-grams

The first retrieval technique was introduced in a previous study [1]. It is based on the vector space model (VSM), which defines a space in which both documents and queries are represented by vectors. Given a query Q and a document D, two vectors q and d are generated. Each component of q and d is a weight associated to a particular indexing term t:

$$q(t)=\begin{cases}1 & \text{if } t\in Q\\ 0 & \text{otherwise}\end{cases} \quad \text{and} \quad d(t)=\begin{cases}1 & \text{if } t\in D\\ 0 & \text{otherwise}\end{cases}. \qquad (1)$$

In our case, the indexing terms t are phone 3-grams [6], i.e. sequences of 3 successive phones extracted from the document and query transcriptions.

This method uses the phone confusion probabilities to compensate for the recognition errors. Let $P(\varphi_1|\varphi_2)$ denote the probability that phone φ_1 is recognized instead of φ_2. A term $P(\varphi|\varphi)$ is the probability that phone φ is correctly recognized. The probability of confusion between two 3-grams t and u is roughly estimated by:

$$P(u|t)=\prod_{i=1}^{3}P(\beta_i|\alpha_i) \ . \qquad (2)$$

where α_i and β_i are the i^{th} phones of t and u respectively. The relevance score proposed in [1] is defined as follows:

$$S_{VSM}(Q,D)=\sum_{t\in Q}P(u_t|t).q(t).d(u_t) \ , \qquad (3)$$

where

$$u_t=\begin{cases}t & \text{if } t\in Q\cap D\\ \arg\left[\max_{t'\in D}P(t'|t)\right] & \text{if } t\in Q\cap\overline{D}\end{cases}. \qquad (4)$$

The terms t that are present in Q but not in D (i.e. $t\in Q\cap\overline{D}$) are approximated by the terms that are the most similar to them in D.

3.3. String matching techniques

The approach proposed in the last section is based on the vector space model. There exists a second, radically different approach to sub-word-based SDR, where the sub-lexical transcriptions of queries and documents are considered as a whole and not as a set of individual terms or sub-sequence units as before. This approach relies on approximate string matching techniques whose goal is to search for approximate occurrences (i.e. taking into account symbol mismatches, insertions, and deletions) of a query string into a document string. A string matching method usually consists of 3 steps:

- *Slot-detection* (SD): As the phone sequences provided by the indexing recognizer do not contain word boundaries, the retrieval system must be able to locate automatically possible occurrences of the query word. A slot detection algorithm is necessary to detect a set of candidate sub-strings (*slots*) within each document transcription for a given query word. A slot is a sub-sequence in which an occurrence of the query string is hypothesized. We used the error-tolerant matching method proposed in [2].
- *Slot-probability-estimation*: This process evaluates how relevant are the detected slots according to the query. A probability is estimated for each slot, according to one of the techniques described below.
- *Relevance score estimation*: For a given document, the relevance score can be obtained by combining the probabilities of the detected slots. In our case, only the best slot is taken into account. The relevance score of a document is simply set to the maximal slot probability.

Two slot-probability-estimation methods were used here: the edit distance (ED) and the probabilistic string matching (PSM) method.

3.3.1. Edit distance

A prerequisite for any string matching search algorithm is the definition of a *distance* (or *similarity measure*) between two given fixed-length sequences of phones. A well-known and

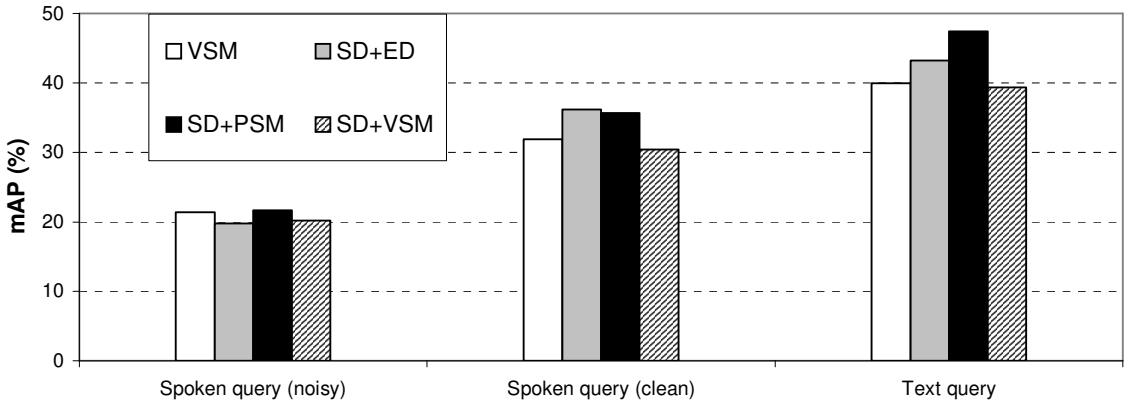

Figure 1: SDR performance measures with 3 different types of queries and 4 retrieval methods.

simple string distance is the *edit distance* (ED). It is defined as the minimum number of operations (insertions, deletions, and substitutions) needed to transform one string (source string) into another (target string). The edit distance is computed by means of a dynamic programming (DP) algorithm [2].

3.3.2. Probabilistic string matching

The edit distance assumes that all phones are separated by an equal acoustic distance of 1. This assumption poorly reflects the real acoustic distance between the slots and the query. In particular, the edit distance approach may be too coarse when strings are corrupted by numerous recognition errors.

The probabilistic string matching (PSM) approach derives the slot-probability more accurately because it takes the probabilities of all types of phone recognition errors (insertions, deletions, and substitutions) into account.

Our PSM algorithm uses the same DP procedure as in the edit distance case. But instead of applying the same uniform cost to every transition in the DP matrix, it associates a different probability of error to each transition. The PSM approach requires more computation effort than the edit distance.

3.4. Combination of slot detection and VSM

Finally, we tested a new combination: the slot scores were evaluated according to equation (3), using the VSM-based approach described in section 3.2. In this case, 3-grams were not extracted from the whole document, but from the detected slots only.

As the VSM method allows the partial matching of phonetic sequences, we assumed that it could improve the retrieval performance in some cases.

4. Experiments

4.1. Database

Experiments have been conducted with data from the PhonDat corpora (PhonDat 1 and 2) consisting of short sentences read by more than 200 German speakers. We built a database of 19306 spoken documents (discarding short utterances of alphanumerical characters) that we indexed as described in section 2. The average length of the documents is 3.9 seconds (37.7 phones per transcription on average).

The evaluation queries consist of 10 city names: Augsburg, Dortmund, Frankfurt, Hamburg, Koeln, Muenchen, Oldenburg, Regensburg, Ulm, Wuerzburg. In a word-based indexing approach, one can imagine that these proper names would be out-of-vocabulary words. A set of relevant documents corresponds to each query (between 96 and 528 documents, depending on the query). We used three distinct sets of queries:

- *Text queries*: The 10 city names are phonetically transcribed via a German pronunciation dictionary [7].
- *Clean spoken queries*: Each city name was recorded once in the same conditions as the spoken documents (high quality microphone, no background noise).
- *Noisy spoken queries*: Each city name was recorded once in adverse conditions (low quality microphone, presence of background noise).

Both sets of spoken queries were transcribed by the phone recognizer used to index the documents (see section 2.2).

4.2. Evaluation

Two popular measures for retrieval effectiveness are *Recall* and *Precision*. Given a set of retrieved documents, the recall rate is the fraction of relevant documents in the whole database that have been retrieved. The precision rate is the fraction of retrieved documents that are relevant.

The precision and recall rates depend on how many documents are kept to form the *n*-best retrieved document list. They vary with *n*, generally inversely with each other. To evaluate the ranked list, a common approach is to plot precision against recall after each retrieved document. We used the plot normalization proposed by TREC [8].

Finally, we evaluate the retrieval performance by means of a single performance measure, called *mean average precision* (mAP), which is the average of precision values across all recall points. It can be interpreted as the area under the Precision-Recall curve. A perfect retrieval system would result in a mean average precision of 100% (mAP = 1).

4.3. Results

In this section we compare the best performing VSM-based method experimented in [1] with methods using slot

detection and string matching techniques. Tests are performed using the three sets of queries separately: noisy spoken queries, clean spoken queries and text queries.

Figure 1 shows the mAP values (reported in Table 1) obtained with different query sets and four retrieval methods:

- SVM: Methods using a vector space model with 3-grams as indexing terms (see section 3.2).
- SD+ED: Slot detection and edit distance matching (see sections 3.3 and 3.3.1).
- SD+PSM: Slot detection and probabilistic string matching (see sections 3.3 and 3.3.2).
- SD+VSM: Slot detection and relevance score estimated with the VSM method (see section 3.4).

Table 1: mAP values (%) obtained with different retrieval methods and different query sets.

	Spoken queries (Noisy)	**Spoken queries (Clean)**	**Text queries**
VSM	21.4	31.9	39.9
SD + ED	19.8	**36.2**	43.2
SD + PSM	21.6	35.7	**47.4**
SD + VSM	20.2	30.4	39.3

4.3.1. Performance with different query sets

As expected, clean spoken queries do not perform as well as text queries. The ASR processing of spoken queries introduces phone recognition errors in the query transcriptions, making the retrieval task more difficult.

Not surprisingly, the performance drops with noisy spoken queries. These queries are recorded in adverse conditions, resulting in even more recognition errors.

4.3.2. Performance with different retrieval methods

The two string matching approaches improve significantly the retrieval efficiency in comparison to the VSM approach, for both text and clean speech queries.

The SD+PSM method yields the best performance with text queries (mAP=47.4%). With clean speech queries, the SD+ED method performs slightly better (mAP=36.2%). The reason may be that the documents and the clean spoken queries were recorded in the same conditions, resulting in phone recognition errors of same nature. In this case, the robustness of the SD+PSM method to recognition errors may not be as determining as before.

With noisy spoken queries, the performance drops significantly (around mAP=20%) for all retrieval methods.

Finally, the SD+VSM approach did not result in any improvement. The introduction of a slot detection step does not improve the performance of the VSM method. The mAP values obtained with VSM and SD+VSM are nearly the same, whatever the query type.

5. Conclusions

The phone-based SDR methods can be roughly classified in two main categories: the methods based on the classical vector space model and the methods using string matching techniques. This paper compared the performance of a VSM-based method with two string matching methods on a particular spoken document retrieval task. A fourth method, combining the slot detection technique with the VSM approach did not yield any improvement.

Our experiments showed that the string matching SDR methods (with slot detection) prove significantly more effective than retrieval based on phoneme n-grams. However, a major weakness of the string matching SDR approaches in general remains the computational cost of the slot detection and string matching algorithms.

Another important point was the evaluation of spoken queries. They do not perform as well as text queries. None of the methods experimented here could compensate efficiently for the recognition errors introduced by the ASR processing of spoken queries. This should be further experimented and improved, for the use of spoken queries makes the system totally independent from any pronunciation vocabulary.

6. References

[1] Moreau N., Kim H.-G. and Sikora T. "Phonetic Confusion Based Document Expansion for Spoken Document Retrieval", *ICSLP Interspeech 2004*, Jeju Island, Korea, October 2004.

[2] Wechsler M., Munteanu E. & Schäuble P., "New Techniques for Open-Vocabulary Spoken Document Retrieval", *SIGIR'98*, pp. 20-27, August 1998.

[3] James D. A., "*The Application of Classical Information Retrieval Techniques to Spoken Documents*", PhD Thesis, University of Cambridge, February 1995.

[4] Ng K. & Zue V. W., "Subword-based Approaches for Spoken Document Retrieval", *Speech Communication*, vol. 32, no. 3, pp. 157-186, October 2000.

[5] Larson M. & Eickeler S., "Using Syllable-based Indexing Features and Language Models to improve German Spoken Document Retrieval", *Eurospeech'03*, pp. 1217-1220, September 2003.

[6] Moreau N., Kim H.-G., Sikora T., "Combination of Phone N-Grams for a MPEG-7-based Spoken Document Retrieval System", to be published in *EUSIPCO 2004*.

[7] Bonn Machine-Readable Pronunciation Dictionary (BOMP): www.ikp.uni-bonn.de/dt/forsch/phonetik/bomp

[8] TREC, "Common Evaluation Measures", *10th Text Retrieval Conference*, pp. A-14, November 2001.