A Decision Tree-Based Method for Speech Processing: Question Sentence Detection

Vũ Minh Quang, Eric Castelli, and Phạm Ngọc Yến

International research center MICA
IP Hanoi – CNRS/UMI-2954 – INP Grenoble
1,Dai Co Viet - Hanoi – Viet Nam
{minh-quang.vu, eric.castelli, ngoc-yen.pham}@mica.edu.vn

Abstract. Retrieving pertinent parts of a meeting or a conversation recording can help for automatic summarization or indexing of the document. In this paper, we deal with an original task, almost never presented in the literature, which consists in automatically extracting questions utterances from a recording. In a first step, we have tried to develop and evaluate a question extraction system which uses only acoustic parameters and does not need any textual information from a speech-to-text automatic recognition system (called ASR system for Automatic Speech Recognition in the speech processing domain) output. The parameters used are extracted from the intonation curve of the speech utterance and the classifier is a decision tree. Our first experiments on French meeting recordings lead to approximately 75% classification rate. An experiment in order to find the best set of acoustic parameters for this task is also presented in this paper. Finally, data analysis and experiments on another French dialog database show the need of using other cues like the lexical information from an ASR output, in order to improve question detection performance on spontaneous speech.

1 Introduction

There're now increasingly important amount of audio data, including voice, music and various kinds of environmental sounds. The audio segmentation and classification are crucial requirements for a robust information extraction on speech corpus. In these recent years, many researches have been conducted in this domain. Lie Lu [1] in his work shows a success automatic audio segmentation and classification into speech, music, environment sound and silence in a two-stage scheme. The first stage of the classification is to separate speech from non-speech, while the second stage further segments nonspeech class into music, environment sounds and silence with a rule based classification scheme. In [9], an online audio classification and segmentation system is presented where audio recordings are also classified and segmented into speech, music, several types of environmental sounds and silence based on audio content analysis. This last system uses many complicated audio features.

In this paper, a new method of speech classification is presented. We plan to classify speech into two classes: question or nonquestion. Different with previous researches which work on audio in general, we treat in our case only one concrete

L. Wang et al. (Eds.): FSKD 2006, LNAI 4223, pp. 1205 – 1212, 2006.

type of audio: speech. We hope that results of this study will be applied in other researches in DELOC project, or in audio browsing, summarization applications developed in our laboratory. Also, our classification is based more importantly on prosodic characteristic than acoustic characteristic of the speech since we use in this experimentation only one prosodic parameter: it's the F0 (which is the fundamental frequency of a speech signal; it's the inverse of the pitch period length. The pitch period is the smallest repeating unit in this speech signal).

The rest of the paper is organized as follows: speech corpus is presented in section II. The classification scheme is discussed in section III. In Section IV, experiment-tation results of proposed method are given while section V concludes our work.

2 Speech Corpus: Telephone Meeting Recording

Our telephone meeting corpus (called Deloc for delocalized meetings) is made up of 13 meetings of 15 to 60 minutes, involving 3 to 5 speakers (spontaneous speech). The total duration is around 7 hours and the language is French. Different types of meetings were collected which correspond to three categories: recruitment interviews; project discussions in a research team; and brainstorm-style talking. This corpus was manually transcribed in dialog acts. For this experimentation of automatic question detection, we have manually selected and extracted a subset of this corpus. It is composed of 852 sentences: 295 questions (Q) sentences and 557 non-questions (NQ) sentences.

3 Automatic Question Extraction System

3.1 Global System Design

The design of our classification system is illustrated in the following figure 1. Beginning by the F0 calculation for all wave files in our corpus, then for characterizing a wave file, 12 features derived from F0 are calculated for each file. Next, all these features among with the name of corresponding wave file are gathered into a database for facility other management. Then, we apply the 10-folds cross-reference testing method by dividing the corpus randomly into 2 subsets: one subset for training decision tree, another subset for testing the decision tree constructed. This step is repeated 10 times in order to find out the best tree with the best classification performance. These processes are presented in more detail in these following sections.

For this meeting corpus (Deloc), we use 200 Q-utterances and 200 NQ-utterances for training and the remaining utterances for testing (95Q + 357NQ).

For the training data, a decision tree is constructed (the decision-tree algorithm used in our experiments is called "C4.5"[1]) and the classifier obtained is evaluated on the remaining test data.

[1] Ross Quinlan, 1993 C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, San Mateo, CA.

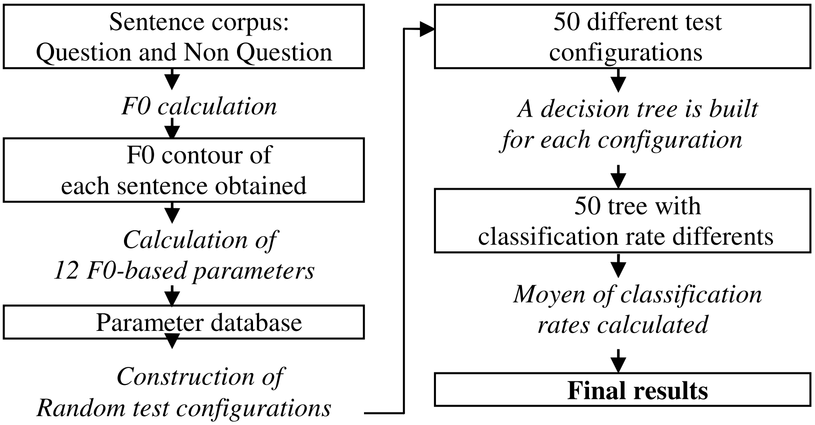

Fig. 1. Global classification system design

The evaluation is either based on a confusion matrix between questions (Q) and non-question (NQ) classes, or on measures coming from information retrieval domain like recall (R), precision (P) and Fratio where :

$$R=\frac{N_{\text{correctly detected questions}}}{N_{\text{total questions in the test set}}} \quad P=\frac{N_{\text{correctly detected questions}}}{N_{\text{total questions detected}}} \qquad FRatio=\frac{2P\cdot R}{P+R}$$

3.2 Feature Vectors

In French language, the interrogative form of a sentence is strongly related to its intonation curve. Therefore, we decided to use the evolution of the fundamental frequency to automatically detect questions on an audio input.

From this F0 curve, we derive a set of features which aim at describing the shape of the intonation curve. Some of these features may be found redundant or basic by the reader: however, our methodology is to first evaluate a set of parameters chosen without too much a priori on the intonation curve for both Q and NQ classes. Then, a detailed discussion concerning the usefulness of each parameter will be provided in the experiments of section 4.2. The parameters defined for our work are listed in Table 1. It is important to note here that, contrarily to classical short term feature extraction procedures generally used in speech recognition, a unique long term feature vector is automatically extracted for each utterance of the database.

These features can be divided into 2 main categories: the first 5 features are the statistics on F0 values, but the 7 next features describe the contour of F0 (raising or falling). The F0 contour was extracted using Praat[2] software.

[2] http://www.fon.hum.uva.nl/praat/

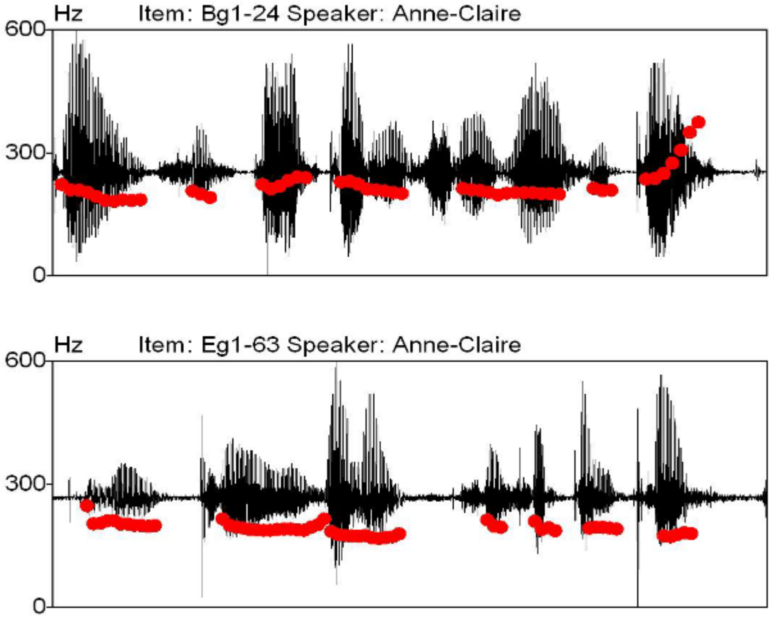

Fig. 2. Signal and F0 contour of French sentences: on top: interrogative sentence; on bottom: affirmative sentence

Table 1. 12-dimensional feature vector derived from F0-curve for each utterance

No	Parameter	Description
1	Min	Minimal value of F0
2	Max	Maximal value of F0
3	Range	Range of F0-values of the whole sentence (Max-Min)
4	Mean	Mean value of F0
5	Median	Median value of F0
6	HighGreaterThan Low	Is sum of F0 values in first half-length smaller than sum of F0 values in last half-length of utterance?
7	RaisingSum	Sum of F0i+1– F0i if F0i+1 > F0i
8	RaisingCount	How many F0i+1 > F0i
9	FallingSum	Sum of F0i+1– F0i if F0i+1 < F0i
10	FallingCount	How many F0i+1 < F0i
11	IsRaising	Is F0 contour rising? (yes/no). Test whether RaisingSum > FallingSum
12	NonZero-FrameCount	How many non-zero F0 values?

3.3 Decision Tree-Based Classifier

Traditionally, statistical-based methods such as Hidden Markov Model (HMM) or Gaussian Mixture Model (GMM) can be used to solve classification problems in speech processing. These statistical methods generally apply on short term features, extracted for instance at a 10ms frame rate. However, in our case, statistical methods are hard to use since we do not use short term feature vectors, as explained in the previous section: one feature vector only is extracted for the whole utterance to be classified, which excludes the use of conventional statistical classifiers. Thus, decision trees, which correspond to another classical machine learning (ML) method [6,7], are a good alternative. In that case, the process is classically divided into two separated stages: training and testing. The training stage is to build a tree-model to represent a set of known-instances while testing stage involves the use of this model to evaluate other unknown-instances. Decision tree is a divide-and-conquer approach to the problem of learning from a set of independent examples (a concrete example is called instance). Nodes in a decision tree involve testing a particular condition, which usually compares an attribute value with a constant. Some other trees compare two attributes with each other, or utilize some functions of one or more attributes. Leaf nodes give a classification for all instances that satisfy all conditions leading to this leaf, or a set of classifications, or a probability distribution over all possible classifications. To classify an unknown instance, it is routed down the tree according to the values of attributes tested in successive nodes, until it reaches a leaf. The instance is classified according to the class assigned to this leaf.

For this work, we have used an implementation of decision-tree algorithms that is included in the open-source toolkit Weka[3] which is a collection of algorithm implementations written in Java for data mining tasks such as classification, regression, clustering, and association rules.

4 Experiments and Results

In this experimentation, we have applied a classic 50-fold cross validation protocol in order to increase the total number of tests made. We repeated 50 times the process of dividing randomly the whole corpus into 2 parts (200Q+200NQ for training; the rest 95Q+357NQ for test). We then obtained 50 decision trees, each with a different classification performance. From these results, we have calculated the mean performance values. Note that the process of training and testing is very fast with decision trees.

Table 2 shows the confusion matrix for Q/NQ classification. The figures correspond to the average performance over all 50 cross-validation configurations. The mean good classification rate is around 75%.

Table 2. Confusion matrix on test data: mean values

Question	Non Question	←classified as
73(77%)	22(23%)	Question
93(26%)	264(74%)	Non Question

[3] http://www.cs.waikato.ac.nz/~ml/weka/

However, it is also interesting to evaluate our question extraction system in terms of precision / recall figures. This is shown in Table 3 where the figures correspond to the average performance over all 50 cross-validation configurations (we give also standard deviations). These results show that our system lead to an acceptable recall (77% of the Q-utterances were retrieved by the system) while the precision is lower, due to a relatively large part of NQ-utterances classified as questions. Looking at the standard deviation calculated over 50 cross-validation configurations, we can say that the system presents a correct stability.

Table 3. Mean performance measures (precision, recall, F-ratio) on test data for the Q-detection task.

	Precision	Recall	Fratio
Mean	44%	77%	56%
Standard deviation	4%	7%	3,5%

This experiment gave us an idea of our first system performance but does not help us to know which acoustic feature is useful for question extraction and which is not. Moreover, since some of these features are correlated with each other, one could try to reduce the original feature set. This is the purpose of the next section experiment.

4.1 Features Ordered by Importance

In order to know how strongly a feature contributes to the classification process, we have performed a "leave-one-out" procedure (as done in [8]). Starting with all N=12 features listed in Table 1, the procedure begins by evaluating the performance of each of the N-1 features. The best subset (in terms of Fratio performance on train data calculated by averaging 50 cross-validation configurations) is determined, and the feature not included in this subset is then considered as the worst feature. This feature is eliminated and the procedure restarts with N-1 remaining features. The whole process is repeated until all features are eliminated. The inverse sequence of suppressed features gives us the list of features ranked from the most effective to the less effective (or most important to less important) for the specific Q/NQ classification task.

This "leave-one-out" procedure lead to the following order of importance (from the most important to the least important feature : 1) isRaising 2) Range 3) Min 4) fallingCount 5) highGreaterThanLow 6) raisingCount 7) raisingSum 8) Median 9) Max 10) nonZeroFramesCount 11) Mean 12) fallingSum.

We observe that isRaising and range parameters are the two most important ones to classify an utterance between Q and NQ classes. The isRaising feature is logically important for detecting questions in French since it corresponds to the case where the F0 contour of a sentence is raising. The second important parameter is the range of the F0 values within the whole sentence. If the decision tree makes use of these only two features isRaising and range, the Fratio obtained is already 54% (to be compared with the 56% obtained with the 12 parameters). Our decision tree can be thus simplified and use two rules only. It is however important to note that some features of the initial parameter set (min, max, RaisingSum, FallingSum) still need to be extracted to calculate these two remaining features.

4.2 Comparison with a "Random" System

In order to really understand what is under the performance obtained in the first experiment, we have compared our system performance to a system that gives a classification output in a basic or random way. For this, we have distinguished three types of "basic" systems: (a) one system which always classifies sentences as Q; (b) one system which always classifies sentences as NQ; and (c) one system that classifies sentences as Q or NQ randomly with a 50% probability. With the data set given in this experimentation, we obtain the following table of Fratio performance for these random systems in comparison with our reference system using 12 parameters (these rates are calculated on test data). In any case, we see that our system is significantly better.

Table 4. Fratio for "basic" or "random" systems and for our system on meeting test data

Always Q (a)	Always NQ (b)	Random 50% Q/NQ (c)	**Our system**
35%	0%	29%	**56%**

5 Conclusion

Retrieving pertinent parts of a meeting or a conversation recording can help for automatic summarization or indexing of the document. In this paper, we have dealt with an original task, almost never presented in the literature, which consists in automatically extracting questions utterances from a recording. In a first step, we have tried to develop and evaluate a question extraction system which uses only acoustic parameters and does not need any ASR output. The parameters used are extracted from the intonation curve and the classifier is a decision tree. Our first experiments on French meeting recordings lead to approximately 75% classification rate. An experiment in order to find the best acoustic parameters for this task was also presented in this paper. Finally, data analysis and experiments on another database have shown the need of using other cues like the lexical information from an ASR output or energy, in order to improve question detection performance on spontaneous speech. Moreover, we plan to apply this Q-detection task to a tonal language like Vietnamese to see if the same parameters can be used or not.

References

1. FERRER L., SHRIBERG E., STOLCKE A., "A Prosody-Based Approach to End-of-Utterance Detection That Does Not Require Speech Recognition", *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, vol. I, Hong Kong, 2003, pp. 608-611.
2. SHRIBERG, E., BATES, R. & STOLCKE, A. "A prosody-only decision-tree model for disfluency detection". *Eurospeech 1997.* Rhodes, Greece.

3. STANDFORD V., GAROFOLO J., GALIBERT O., MICHEL M., LAPRUN C., "The NIST Smart Space and Meeting Room Projects: Signal, Acquisition, Annotation and Metrics", *Proc of ICASSP 2003*, Hong-Kong, China, Mai 2003.
4. WANG D., LU L., ZHANG H.J., "Speech Segmentation Without Speech Recognition", *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, vol I, april 2003, pp. 468-471.
5. MANA N., BURGER S., CATTONI R., BESACIER L., MACLAREN V., McDONOUGH J., METZE F., "The NESPOLE! VoIP Multilingual Corpora in Tourism and Medical Domains" *Eurospeech 2003*, Geneva, 1-4 Sept. 2003.
6. MARQUEZ L., “Machine learning and Natural Language processing”, Technical Report LSI-00-45-R, Universitat Politechnica de Catalunya, 2000.
7. WITTEN I.H., FRANK E., *Data mining: Pratical machine learning tools and techniques with Java implementations*, Morgan Kaufmann, 1999.
8. L. BESACIER, J.F. BONASTRE, C. FREDOUILLE, “Localization and selection of speaker-specific information with statistical modeling". *Speech Communication, n°31 (2000),* pp 89-106