# Extraction automatique de Questions dans les corpus de réunions et de dialogues

Minh Quang Vu, Laurent Besacier, Eric Castelli, Ngoc Yen Pham

# Extraction automatique de Questions dans les corpus de réunions et de dialogues

**Vũ Minh Quang[1,2] , Laurent Besacier[1] , Eric Castelli[2], Phạm Ngọc Yến[2]**

[1]Laboratoire CLIPS/IMAG,
Université Joseph Fourier
385, rue de la Bibliothèque
BP53, 38041 Grenoble cedex 9, France
{quang.vu-minh,laurent.besacier}@imag.fr

[2]Centre de recherche international MICA
Institut Polytechnique de Hanoi
1,Dai Co Viet
Hanoi – Viet Nam
{eric.castelli,ngoc-yen.pham}@mica.edu.vn

**Résumé :** L'extraction de parties pertinentes d'un enregistrement de parole d'une réunion ou d'une conversation peut aider à réaliser le résumé automatique ou l'indexation du document audio ou audio-vidéo. Nous présentons un travail original, peu étudié dans la littérature, qui porte sur l'extraction automatique de phrases de type questions à partir d'un enregistrement audio. Dans une première démarche, nous avons développé et évalué un système d'extraction de questions qui utilise seulement des paramètres acoustiques mesurés sur le signal de parole sans utiliser les résultats d'un module de reconnaissance RAP. Les paramètres utilisés sont extraits de la courbe d'intonation et le classificateur est un arbre de décision. Nos premières expérimentations sur un corpus français de réunions nous ont permis d'obtenir un taux de classification de 75 % environ. Une deuxième phase de l'étude a été menée pour trouver le meilleur jeu de paramètres acoustiques pour cette tâche. Nous avons alors appliqué notre système sur un autre type de corpus (dialogues en langue française), ce qui a démontré que les seuls paramètres acoustiques ne sont pas suffisants et qu'il semble nécessaire d'utiliser d'autres indices, comme l'information lexicale de la sortie d'un moteur de RAP, pour améliorer la performance de détection de questions dans le discours spontané.

**Mots Clés :** Apprentissage ; Systèmes de dialogues ; Traitement automatique des langages.

## 1 INTRODUCTION

Face au volume croissant des données audio disponibles, les systèmes d'indexation et de classification deviennent indispensables, afin de pouvoir localiser le plus rapidement possible les enregistrements désirés. Ces dernières années, de nombreuses études ont vu le jour dans ce domaine. Lie Lu [LU 2001] a démontré avec succès que son système est capable de classifier un flux audio en parole, musique, bruit de fond et silence par un processus à deux phases : la première phase de classification consiste à séparer parole/non parole, alors que la deuxième phase discrimine ensuite le signal audio en musique, bruit de fond et silence avec un classifieur par règles. Un autre système [ZHA 1998] utilise de nombreux paramètres complexes pour classifier et segmenter du signal audio en parole, musique, quelques types de bruits environnementaux et silence. Ces systèmes utilisent usuellement des méthodes de classification comme GMM (Gaussian Mixture Model), BP-ANN (Back Propagation Artificial Neural Network) et KNN (K-Nearest Neighbor). Dans cette étude, nous proposons un nouveau système de classification basé sur l'utilisation d'un arbre de décision. Le but est de classifier le signal de parole en deux catégories : *question* et *non question*. A la différence des travaux qui manipulent le signal audio d'une manière globale, nous allons traiter dans notre expérimentation un seul type de signal : la parole. Nos résultats obtenus pourraient être appliqués à des domaines tels que la gestion de documents sonores, la réalisation automatique de résumés de discours ou de réunions, la recherche d'informations, parce que les segments de parole autour d'une question contiennent généralement des informations pouvant s'avérer très utiles dans ces applications.

Cet article est composé comme suit : le corpus est présenté dans la section 2, les détails du système de classification sont présentés section 3, et enfin, les sections 4 et 5 présentent respectivement les résultats obtenus et la conclusion du travail.

## 2 LE CORPUS DELOC

Nous utilisons le corpus du projet DELOC mené dans notre laboratoire, dont le but consistait à étudier différents types de réunions, ainsi que les différentes façons de s'exprimer (comportements langagiers selon les types de réunions). Le but du projet est de proposer

des outils « collaboratifs » associés à la visioconférence, ou à n'importe quel contexte de réunions délocalisées, c'est-à-dire des outils d'aide à la rédaction du compte rendu en fin de réunion, ou d'aide à la transcription.
Ce corpus se compose de différents types de réunions délocalisées réalisées par téléphone : 1) « brainstorming » ou remue-méninges, 2) (pré-)entretien d'embauche, 3) réunion de projet. Ces enregistrements ont été segmentés manuellement en phrases qui correspondent chacune à une *question* ou une *non question* : au total 852 phrases dont 295 phrases *questions* et 557 phrases *non questions*. Les phrases de courte durée correspondent à : « *Allo?* », « *D'accord* »…alors que celles à durée longue correspondent par exemple à : « *parce que chez Multicom, j'imagine qu'il y a quand même...il y a quand même des gens qui pourraient peut être compléter ?* ». Notre système peut traiter non seulement des phrases à rythme normal, mais encore celles dont le rythme est hésitant

# 3 LE SYSTEME DE CLASSIFICATION EN *QUESTION* ET *NON QUESTION*

## 3.1 Structure globale du système

La structure globale du système de classification est illustrée à la figure 1.

La classification commence par le calcul, pour toutes les phrases du corpus, de la fréquence fondamentale (F0 ou intonation) du signal de parole parce que la variation de l'intonation est l'une des caractéristiques principales qui différencient les types de phrases parlées. Puis, dans le but de caractériser cette intonation par un ensemble de paramètres, 12 paramètres dérivés des valeurs instantanées de F0 sont calculés. Nous construisons alors, afin de faciliter la gestion, une base de données qui comprend toutes les phrases et, pour chacune d'entre elles, ses paramètres associés

## 3.2 Les paramètres de caractérisation utilisés

A la différence des travaux récents dans le domaine [LU 2001, FER 2003, WAN 2003] qui utilisent des vecteurs sophistiqués de paramètres acoustiques tels que *la fréquence fondamentale (F0)*, *le taux de passage par zéro (zero-crossing rate ZCR)*, *le rapport d'énergie à court terme (low short-time energy ratio LSTER)*, *le flux spectral (spectrum flux SF)*, nous utilisons dans notre expérimentation uniquement le paramètre F0 qui est calculé directement à partir du signal, en découpant celui-ci en fenêtres de 20ms. Cependant, à partir de F0, d'autres paramètres peuvent en être dérivés et nous proposons un ensemble de 12 paramètres listés dans le tableau 1.

Nous pouvons remarquer que ces paramètres peuvent se diviser en deux catégories distinctes : les 6 premiers paramètres sont des statistiques sur la valeur de F0, alors que les 6 derniers caractérisent le contour (la forme) de l'évolution de F0 (contour montant ou descendant). L'utilisation de ces 6 derniers paramètres constitue l'originalité de notre méthode, au niveau de la caractérisation du signal de parole

## 3.3 Méthode de classification par l'arbre de décision

Traditionnellement, les méthodes statistiques telles que les modèles de Markov cachés *(HMM)* ou les modèles de mixtures de Gaussiennes *(GMM)* et leurs variantes sont utilisées en Traitement Automatique des Langues Naturelles (TALN). L'arbre de décision est une méthode classique d'apprentissage [FRA 1999] s'utilisant de manière similaire : le processus entier se compose de deux phases séparées, apprentissage et test.

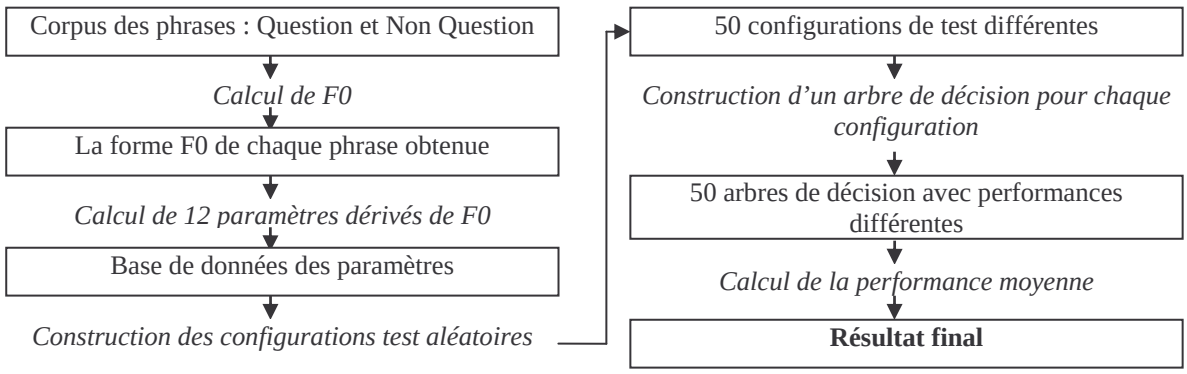


**Figure 1.** Structure globale du système de classification

| No | Paramètre | Description |
|---|---|---|
| 1 | Min | Valeur minimale de F0 |
| 2 | Max | Valeur maximale de F0 |
| 3 | Range | Gamme de F0 pour la phrase entière (Max-Min) |
| 4 | HighGreaterThanLow | Est-ce que la somme des valeurs F0 dans la première moitié de la phrase est supérieure à celle des valeurs F0 dans la dernière moitié ? |
| 5 | Mean | Moyenne des valeurs de F0 d'une phrase |
| 6 | Median | Médiane des valeurs F0 d'une phrase |
| 7 | RaisingSum | Somme des F0i+1 − F0i si F0i+1 > F0i |
| 8 | RaisingCount | Combien de F0i+1 > F0i |
| 9 | FallingSum | Somme des F0i+1 − F0i si F0i+1 < F0i |
| 10 | FallingCount | Combien de F0i+1 < F0i |
| 11 | IsRaising | Est-ce que la forme F0 est montante ? (oui/non) Teste si RaisingSum > FallingSum |
| 12 | NonZeroFrameCount | Combien de valeurs de F0 sont non nulles ? |

**Table 1 :** Les 12 paramètres dérivés de F0

L'apprentissage consiste à construire un modèle représentant un ensemble des éléments, alors que le test utilise ce modèle pour évaluer un nouvel élément inconnu. Ces dernières années, beaucoup de travaux en TALN ont adopté des solutions d'apprentissage. Cependant, depuis les années 2000, une tendance vers une utilisation mixte des 2 types d'algorithmes s'affirme [MAR 2000]. L'arbre de décision est une approche

```
IsRaising = yes
| FallingCount <= 53
| | highGreaterthanLow = yes: yes (69.0/7.0)
| | highGreaterthanLow = no
| | | min <= 116.50566
| | | | RaisingSum <= 1197.5662: yes (7.0/2.0)
| | | | RaisingSum > 1197.5662: no (4.0)
| | | min > 116.50566: yes (8.0)
```

**Figure 2. Extrait d'un arbre obtenu. Le yes/no qui se trouve en feuille (après ':') signifie le résultat de classification : IsQuestion=yes ou IsQuestion=no.**

| Question | Non Question | ←classifier comme |
|---|---|---|
| 184(92%) | 16(8%) | Question |
| 27(13%) | 173(87%) | Non Question |

**Tableau 2. Matrice de confusion sur les données d'apprentissage.**

| Question | Non Question | ←classifier comme |
|---|---|---|
| 73(77%) | 22(23%) | Question |
| 93(26%) | 264(74%) | Non Question |

**Tableau 3. Matrice de confusion sur les données de test (valeurs moyennes).**

| | Précision | Rappel | F_ratio |
|---|---|---|---|
| Moyenne | 44,2% | 76,5% | 55,7% |
| Ecart-type | 4% | 7,2% | 3,5% |

**Tableau 4. Les mesures moyennes sur les données test de la classe question.**

*diviser-et-conquérir* pour le problème de l'apprentissage à partir
d'un ensemble d'éléments indépendants (un élément concret est appelé une *instance*).
Un nœud dans l'arbre consiste à tester une condition particulière qui, en général, compare la valeur d'un attribut avec une constante, ou compare ensemble deux attributs, ou encore utilise des fonctions mathématiques d'un ou plusieurs attributs. La feuille d'arbre donne soit une classification des éléments satisfaisant toutes les conditions menant à cette feuille, soit un ensemble de classification, ou soit une distribution probabiliste sur toutes les classifications possibles.
Pour classifier une *instance* inconnue, l'algorithme teste les attributs dans les nœuds jusqu'à ce qu'il atteint une feuille. Là, cette instance est classifiée selon la classe attribuée à la feuille. L'implémentation de l'algorithme provient du logiciel *open-source* Weka (http://www.cs.waikato.ac.nz/~ml/weka/) qui comprend les algorithmes de *classification*, *régression*, *clustering*, *règles d'association* écrits en Java.

## 4 RESULTATS EXPERIMENTAUX

Il y a 295 phrases *question* et 557 phrases *non question* dans le corpus. Nous appliquons la méthode *50-folds cross validation*, c'est-à-dire nous répétons 50 fois le processus de division aléatoire du corpus en deux parties : une pour l'apprentissage (200 *questions* et 200 *non questions*), une pour le test (le reste : 95 *questions* et 357 *non questions*). On obtient alors 50 arbres, chacun présentant une performance différente. Il ne reste plus qu'à calculer la performance moyenne de ces arbres. La construction de l'arbre se fait rapidement, l'évaluation de l'arbre sur les données de test est aussi rapide (en dixièmes de millisecondes). Avec un écart-type de 2,4%, la performance moyenne de classification obtenue est de 84,5% (c'est-à-dire 84,5% des instances sont correctement classifiée). Un exemple d'arbre construit est donné à la figure 2.

Le tableau 2 présente plus en détails la performance de classification sur les données d'apprentissage.
Pour l'évaluation sur les données de test, du fait que les nombres de *questions* et de *non questions* ne sont pas égaux (97 vs 357), nous devons évaluer les notions de *précision*, *rappel*, *F_ratio* de la classe *question* (voir tableau 4), avec la matrice de confusion donné dans le tableau 3. L'indice F_ratio mesurant le taux de bonne classification de la classe *question*, nous permet de conclure que le système est assez performant (55,7%).

## 5 CONCLUSION

Nous avons présenté une nouvelle méthode de classification de parole en *question* et *non question* par l'utilisation d'un arbre de décision. Dans notre expérimentation, un seul paramètre prosodique F0 est calculé directement à partir du signal, et 12 autres paramètres sont dérivés de F0 pour construire l'arbre de décision. Ce résultat peut s'appliquer pour d'autres applications en parole telles que le résumé automatique, la navigation ou la recherche d'information, car les zones autour d'une question contiennent souvent des informations importantes à identifier. Afin d'augmenter la performance du système, davantage de paramètres sont à étudier. D'autres classes avec intonations peuvent aussi être analysées comme les exclamations, les ordres, etc.

## BIBLIOGRAPHIE


[LU 2001] LU L., JIANG, H., ZHANG H.J., "A Robust Audio Classification and Segmentation Method", *9th ACM Int. Conf. on Multimedia*, 2001, pp.203-211.

[ZHA 1998] ZHANG T., KUO C.C.J., "Content Based Classification and Retrieval of Audio", *SPIE's 43rd Ann. Meeting-Conf. on Advanced Signal Processing Algorithms, Architectures and Implementations VII*, SPIE Vol. 3461, San Diego, july 1998, pp. 432-443.

[FER 2003] 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.

[WAN 2003] 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,.

[FRA 1999] WITTEN I.H., FRANK E., *Data mining: Pratical machine learning tools and techniques with Java implementations*, Morgan Kaufmann, 1999.

[MAR 2000] MARQUEZ L., Machine learning and Natural Language Processing, Technical Report LSI-00-45-R, Universitat Politechnica de Catalunya, 2000.