



Master 2 ATAL

Applications Multilingues

---

# Segmentation en mots du Japonais

---

*Étudiant :*  
Coraline MARIE

*Encadrant :*  
Florian BOUDIN

3 novembre 2014



UNIVERSITÉ DE NANTES

## Table des matières

<b>Introduction</b>	<b>2</b>
<b>1 Première approche</b>	<b>2</b>
1.1 La segmentation par les Modèles de Markov cachés . . . . .	2
1.2 Gestion des probabilités . . . . .	2
<b>2 Implémentation des trigrammes</b>	<b>3</b>
<b>3 Reconnaissance des alphabets</b>	<b>3</b>
<b>Conclusion</b>	<b>3</b>

## Introduction

Le Japonais est une langue très particulière, dont l'écriture diffère totalement des langues latines. L'utilisation d'alphabets différents et l'absence de marqueurs explicites, tel que les espaces, la rend plus délicate à traiter, notamment dans le cadre de la Recherche d'Information.

Afin de faciliter le traitement de textes en langue Japonaise, il faut au préalable appliquer un processus de segmentation en mots, appelée *tokenisation*. Ce rapport présente donc différentes méthodes combinables permettant de segmenter en mots, des textes en japonais. Il y aura d'abord une description de l'implémentation de la méthode des Modèles de Markov cachés de Constantine P. Papageorgiou[1]. Il y aura également une présentation de l'amélioration de cette méthode par l'implémentation de ngrammes, avant de finir par l'explication d'une méthode reposant sur l'analyse des différents alphabets utilisés par la langue Japonaise.

## 1 Première approche

### 1.1 La segmentation par les Modèles de Markov cachés

L'article *Japanese word segmentation by hidden markov model* de Constantine P. Papageorgiou[1] présente une méthode de segmentation en mots du Japonais très efficace. Cette méthode plutôt simple, utilise à la fois un corpus d'entraînement déjà segmenté et un second corpus qu'il faudra segmenter.

La première étape de cette méthode consiste à analyser bigramme par bigramme tout le corpus d'entraînement, afin de mémoriser le comportement de chaque duo de caractères : *coupure* ou *non-coupure*. Cette analyse permet ensuite d'obtenir des probabilités de comportement sur l'ensemble des bigrammes rencontrés.

La seconde étape consiste à analyser bigramme par bigramme tout le corpus à segmenter, puis par l'intermédiaire d'un *HMM* (Hidden Markov Model : Modèle de Markov caché), elle définit s'il faut couper ou non le bigramme.

Les résultats de l'implémentation de cette méthode sur le corpus de test donnent un peu plus de 89,2% de f-mesure :

Avg Precision	0.904005681695
Avg Recall	0.881382517888
Avg f-measure	0.892550767507

### 1.2 Gestion des probabilités

La première limite qui est observable sur cette méthode est la gestion pauvre des probabilités non observés. En effet, dans le cas où un bigramme du corpus à segmenter n'a pas été observé dans le corpus d'entraînement, la méthode attribue la même probabilité à la coupure et à la non coupure du bigramme.

Or, si on regarde plus attentivement les probabilités de coupure des bigrammes dans le corpus d'entraînement, on remarque qu'il est plus probable de couper que de ne pas couper un bigramme.

Ainsi, la première amélioration faite à cette méthode est l'attribution de deux probabilités différentes à la coupure et à la non-coupure pour les bigrammes non rencontrés. Ces probabilités sont 0.02 pour la coupure et 0.01 pour la non-coupure. Elles ont été définies pendant la phase d'entraînement, par décompte et normalisation du nombre total de bigrammes rencontrés dans le corpus d'entraînement.

Les résultats de cette première amélioration donnent environ 92% de f-mesure :

Avg Precision	0.87553154073
Avg Recall	0.965715790723
Avg f-measure	0.918415054529

## 2 Implémentation des trigrammes

## 3 Reconnaissance des alphabets

Comme il l'a été dit précédemment, le Japonais est composé de plusieurs alphabet :

## Conclusion

Bien que les scores obtenus grâce à ses diverses méthodes soient satisfaisants (94.6% de f-mesure sur le corpus de test), il reste encore des erreurs. Ceci peut être rapproché au corpus d'entraînement dont la taille reste limitée, mais pas seulement.

Il y a eu pendant ce projet d'autres idées qui n'ont pas eu le temps d'être implémentées, comme par exemple :

- le traitement des alphabets par ngrammes ;
- l'ajout d'un dictionnaire : ce qui devrait éliminer les groupes de caractères inexistants ;
- l'entraînement du modèle de Markov caché sur d'autres types de corpus tokenisés :
  - un corpus web japonais (ex : Wikipédia) avec récupération des *liens* : ce qui permettrait de mettre en évidence des mots tokenisés dans leur contexte ;
  - un corpus journalistique (ex : blogs de presse, journaux, ...) avec reconnaissance de formes et de mise en page (titres, texte en italique, texte en gras, ...) ce qui pourrait également aider à la reconnaissance des mots tokenisés.

## Références

- [1] Constantine P. Papageorgiou. Japanese word segmentation by hidden markov model. *Proceedings of the workshop on Human Language Technology*, HLT '94, 1994.