Reconnaissance automatique de la parole en langue khmère : quelles unités pour la modélisation du langage et la modélisation acoustique?

Sopheap Seng, Sethserey Sam, Viet-Bac Le, Brigitte Bigi, Laurent Besacier

LIG, UMR CNRS 5217
BP 53, 38041 Grenoble Cedex 9, FRANCE
e-mail : Sopheap.Seng@imag.fr

ABSTRACT

In this paper we present an overview on the development of a large vocabulary continuous speech recognition system for Khmer language. Methods and tools used for language resources collection for quick development of an ASR system for a new under-resourced language are presented. Face with the problem of lack of text data and the word error segmentation in language modeling, we investigate how different views of the text data (word and sub-word units) can be exploited for Khmer language modeling. We propose to work both at the model level (by making hybrid vocabularies with both word and sub-word units) as well as at the ASR output level (systems combination). For acoustic modeling, we use basic linguistic rules to automatically generate pronunciation dictionaries based on grapheme or phoneme. An experimental framework is setup to evaluate the performance of each modeling units.

Keywords: ASR, Khmer, word and sub-word units language modeling, acoustic modeling

1. INTRODUCTION

Le développement d'un système de Reconnaissance Automatique de la Parole continue grand vocabulaire (RAP) pour une langue peu dotée comme le khmer est une tâche qui conduit à trois challenges : (1) le manque de ressources linguistiques sous forme numérique (corpus de texte et de parole), (2) le système d'écriture sans séparation explicite entre les mots, qui nécessite une segmentation automatique pour que la modélisation statistique de langage soit applicable et (3) les caractéristiques acoustiques et phonologiques de la langue qui sont encore assez peu connues.

Cet article présente une vue d'ensemble sur le développement d'un système RAP pour le khmer. Nous décrivons tout d'abord notre méthode de collecte des données linguistiques pour le développement rapide d'un nouveau système de RAP pour une langue peu dotée. Le problème du manque de données textuelles et de la présence des erreurs lors de la segmentation en mots est abordé. Nous essayons de traiter ce problème en exploitant plusieurs vues sur les données textuelles dans la modélisation du langage. Nous travaillons à la fois au niveau du modèle en créant des vocabulaires hybrides à partir d'unités lexicales et sous-lexicales, et en combinant des sorties de différents systèmes de RAP pour décoder une meilleure hypothèse. Pour la modélisation acoustique, nous présentons et comparons des méthodes de génération automatique de dictionnaires de prononciation à base de graphèmes et à base de phonèmes pour le khmer. Enfin, des expérimentations sont menées pour tester et comparer les approches proposées.

2. ACQUISITION DE DONNEES LINGUISTIQUES

2.1. Acquisition du corpus de texte

Une grande quantité de données textuelles (plusieurs centaines de millions de mots) est nécessaire pour obtenir une estimation précise des probabilités des n-grammes d'un modèle de langage. Collecter des textes à partir du web est devenu une approche standard, car ceci permet d'obtenir gratuitement et rapidement une grande quantité de textes. Dans [1], un robot explore le web et extrait les textes pour construire un corpus. Une autre approche [2] consiste à estimer les probabilités des n-grammes directement à partir du web en utilisant les statistiques données par un moteur de recherche. Ces méthodes s'appliquent bien aux langues comme le français ou l'anglais qui disposent d'une grande couverture sur l'Internet. Cependant, les problèmes pour les langues peu dotées comme le khmer, concernent le nombre limité de sites web, la faible vitesse de transmission et la qualité variable des documents qui nécessitera alors plus d'outils de traitement. On préférera par exemple des sites de nouvelles en khmer, au fort contenu rédactionnel au lieu de parcourir tous les sites web contenant très peu de données exploitables.

Une fois les pages html récupérées, de nombreux traitements sont nécessaires afin de construire un corpus de texte : (1) extraction du contenu textuel, (2) conversion des encodages (de l'encodage ad-hoc non standardisé vers Unicode), (3) segmentation du texte en phrases et en différentes unités lexicales et/ou sous-lexicales, (4) transcription des caractères spéciaux, des nombres et normalisation des orthographes.

Nous avons adapté le toolkit ClipsTextTk [3], développé pour le français et pour le vietnamien, en y ajoutant les traitements spécifiques à la langue khmère. Nous avons introduit des outils pour la conversion de l'encodage, la transcription des caractères spéciaux et des nombres. Pour la segmentation automatique, nous avons développé les outils de segmentation en phrase, en mot, en syllabe et en clusters de caractères (comme présenté en table 1). La segmentation en mots repose sur un algorithme « plus longue chaîne d'abord » (*Longest Matching*), en utilisant

un vocabulaire de 18000 mots du dictionnaire officiel *Chhoun Nat*. La performance obtenue est de 95 %. La segmentation en syllabes et en clusters de caractères est basée sur des règles linguistiques. La segmentation en syllabes, qui n'est pas triviale, ne donne qu'une performance de 85 % tandis que la segmentation en clusters de caractères est 100% correcte. La structure d'un cluster de caractères khmer est, en effet, non ambiguë et atomique. 25130 pages html en langue khmère ont été collectées à partir de 5 sites de nouvelles (soit 448 Mo). Après normalisation, le corpus de texte est constitué de 0,5 millions de phrases, et 15,5 millions de mots.

Table 1: Exemple de segmentation d'une phrase khmère

Phrase	ព្រះពុទ្ធជាព្រះបរមគ្រូនៃយើង												
Mot	ព្រះពុទ្ធ			ជា	ព្រះបរម				គ្រូ	នៃ	យើង		
Syllabe	ព្រះ		ពុទ្ធ		ជា	ព្រះ		ប	រម	គ្រូ	នៃ	យើង	
CC	ព្រះ		ពុ	ទ្ធ	ជា	ព្រះ		ប	រ	ម	គ្រូ	នៃ	យើ	ង
Traduction	Le bouddha est notre grand maître												

2.2. Acquisition des signaux de parole

Le recueil de signaux de parole est souvent réalisé en faisant l'enregistrement des textes prononcés par des locuteurs professionnels dans un studio. Cette tâche fastidieuse demande beaucoup de ressources. Pour obtenir rapidement un corpus de parole khmère, nous avons enregistré des émissions de type bulletin d'information des chaînes de radio locales à Phnom Penh, Cambodge, en coopération avec l'Institut de Technologie du Cambodge (ITC). Une campagne de transcription manuelle des signaux a été organisée à l'ITC. 20 étudiants volontaires motivés à contribuer au développement des ressources pour la langue khmère ont été recrutés et formés. En utilisant le logiciel open source *Transcriber* [4], 6h30mn de signaux ont été transcrits à ce jour. Le corpus de parole contient 3200 phrases prononcées par 8 locuteurs (3 femmes). 172 phrases ont été extraites afin de constituer les données de test des expérimentations.

3. MODELISATION DU LANGAGE

La nature statistique des approches utilisées dans la modélisation du langage par n-grammes, nécessite une grande quantité de corpus de textes pour obtenir une estimation précise des probabilités. Le mot, bien souvent définit à tort comme une séquence de caractères séparée par des espaces, est souvent l'unité de base de cette modélisation. Cette approche s'applique bien aux langues comme le français et l'anglais. Pour les langues qui possèdent un système d'écriture sans frontière explicite entre les mots, on doit utiliser un système imparfait de segmentation automatique en mots, qui introduira des erreurs dans l'estimation du vocabulaire et du modèle de langage. De plus, la quantité de données disponible étant limitée, l'estimation des probabilités du modèle de langage n'est pas bonne et cela nuit à la performance du modèle de langage. Une alternative possible consiste à calculer les probabilités à partir d'unités sous-lexicales (syllabes ou clusters de caractères). Ces derniers permettent une estimation des probabilités plus précise, le vocabulaire étant généralement plus petit. En contrepartie, la couverture des n-grammes est plus réduite. Parmi les travaux existants qui utilisent les unités sous- lexicales pour la modélisation du langage, nous pouvons citer [5, 6 et 7] qui utilisent les morphèmes pour l'arabe, le finnois, ou le somalien. Pour une langue non-segmentée comme le japonais, le caractère est utilisé dans [8]. Dans le cas du khmer, le texte peut être segmenté en mots, syllabes ou clusters de caractères. A priori, les clusters de caractères semblent être une bonne unité de modélisation, étant donnée que la segmentation est triviale et sans ambiguïté. Un des objectifs de cet article est d'analyser comment ces différentes vues sur les données textuelles peuvent être exploitées au mieux pour la reconnaissance automatique de la parole khmère.

Au niveau des modèles de langage, l'idée est qu'à partir d'un vocabulaire initial d'unités sous-lexicales (syllabe ou clusters de caractères), nous ajoutons progressivement les N mots les plus fréquents. En augmentant N, nous obtenons plusieurs vocabulaires hybrides mots/syllabes ou mots/clusters de caractères. Ces différents vocabulaires sont utilisés pour entraîner les modèles n-grammes. La performance de ces modèles est donnée dans la section 5 consacrée aux expérimentations.

4. MODELISATION ACOUSTIQUE

4.1. Dictionnaires de prononciation

Le dictionnaire de prononciation fournit le lien entre les séquences des unités acoustiques et les mots représentés dans le modèle de langage. Alors que les corpus de texte et de parole peuvent être collectés, le dictionnaire de prononciation n'est généralement pas directement disponible. Bien qu'un dictionnaire de prononciation créé manuellement donne une bonne performance, la tâche est très lourde à réaliser et demande des connaissances approfondies sur la langue en question. La littérature propose des approches qui permettent de générer automatiquement le dictionnaire de prononciation. L'approche, simple et totalement automatique, qui utilise des graphèmes comme unité de modélisation a été bien validée dans [9, 10].

Table 1: Phonèmes Khmers

Type de phonème		Symbole
Consonne initiale *CI*	*CI* simple	k kʰ ŋ c cʰ ɲ d t tʰ n p pʰ b m r l s h ʋ j ʔ
	CI double	85 *CI* doubles possibles (voir [11] pour une liste complète)
Voyelle	courte	i e j ə ɑ a u o
	longue	iː eː ɛː ɨː əː ɑː aː uː oː ɔː
	diphtongue	iə ei j ə əj ao uə ou ɔ ə eəˇ uəˇ oəˇ
Consonne finale *CF*		k c t p h n ŋ ɲ m j l ʋ ʔ

Pour le khmer, nous utilisons deux méthodes pour générer automatiquement les dictionnaires de prononciation. La première méthode est fondée sur le graphème et consiste à représenter chaque graphème khmer comme une unité de modélisation. La deuxième méthode est fondée sur des règles de conversion de graphème à phonème. Ces règles sont créées à partir des connaissances de la structure des syllabes khmères, de

règles de prononciation et de la liste des phonèmes (tableau 1) décrits dans le manuscrit de Huffman [11].

4.2. Modélisation acoustique

Notre dictionnaire de prononciation à base de graphèmes contient 77 graphèmes utilisés comme unités de modélisation. Dans le cas de la modélisation à base de phonèmes, nous utilisons les phonèmes simples comme unités de base. Un cluster de consonnes est considéré comme une séquence de 2 consonnes simples : */pt/* → */p/ + /t/*. Comme les voyelles longues et courtes possèdent les mêmes propriétés acoustiques mais avec des durées de voisement différentes (les voyelles longues sont en général deux fois plus longues que les voyelles courtes [12]), une voyelle longue est représentée comme la concaténation de deux voyelles courtes : */e:/* → */e/ + /e/*. De la même manière, les diphtongues sont considérées comme une séquence de voyelles simples. Nous avons finalement 33 phonèmes.

Nous utilisons SphinxTrain [13] pour entraîner les modèles acoustiques (HMMs). Des modèles indépendants du contexte (CI) et dépendants du contexte (CD avec 1000 états) à base de graphèmes et de phonèmes sont construits à partir des corpus de parole décrits en section 2.2. Nous obtenons ainsi 4 modèles acoustiques, à savoir *Grapheme_CI, Grapheme_CD, Phoneme_CI* et *Phoneme_CD.*

5. EXPERIMENTATIONS ET RESULTATS

5.1. Système de RAP

Les expérimentations sont menées avec Sphinx3 [13]. La topologie des modèles est un HMM de 3 états avec 8 Gaussiennes par état. Le vecteur de paramètres contient 13 MFCCs, ses premieres et secondes dérivées.

Le corpus de texte est d'abord segmenté en mots et les 20k mots les plus fréquents sont extraits pour servir de vocabulaire de test. Ce vocabulaire de mots et le corpus d'apprentissage des modèles de langage sont ensuite segmentés en 8800 syllabes et 3500 clusters de caractères respectivement. La transcription du corpus d'apprentissage de parole est aussi utilisée pour apprendre le modèle de langage. Les modèles de langage utilisés dans nos expérimentations sont obtenus par l'interpolation linéaire entre les modèles créés à partir des données web et ceux de la transcription de corpus de parole. Des données de développement sont utilisées pour optimiser les paramètres d'interpolation.

En plus du taux d'erreur de mots (WER), nous utilisons systématiquement le taux d'erreur de syllabes (SER) et le taux d'erreur de cluster de caractères (CCER) pour l'évaluation du système de RAP, car la segmentation de mots et syllabes khmers n'est pas triviale et les erreurs de segmentation pourraient empêcher une comparaison correcte entre les systèmes. Les tests sont effectués sur le corpus de test qui contient 172 phrases (environ 20mn de parole).

5.2. Graphème Vs Phonème

Dans cette expérimentation, nous voulons comparer la performance de nos différents modèles acoustiques.

Table 2 : Résultats avec différents modèles acoustiques

Modèle acoustique	WER	SER	CCER
Grapheme_CI	64,9	39,9	33,6
Grapheme_CD	47,8	26,9	20,8
Phoneme_CI	57,9	38,2	31,9
Phoneme_CD	49,6	25,1	19,1

Les résultats de la table 2 montrent que les modèles dépendants du contexte sont meilleurs que les modèles indépendants du contexte, même si la quantité de données d'apprentissage est faible (moins de 7h). Les performances des modèles à base de graphèmes et à base de phonèmes sont très comparables, ce qui montre le potentiel de l'approche à base de graphèmes dans le contexte de cette langue peu dotée. La différence observée entre WER et CCER est partiellement due aux erreurs de segmentation dans les hypothèses et les références. Le CCER semble plus adapté pour les évaluations car il permet des comparaisons plus justes (sans erreurs de segmentation).

5.3. Modèles de langage de mots et sous-mots

Pour observer le potentiel des différentes unités lexicales et sous-lexicales, trois modèles de langage trigrammes sont appris en utilisant respectivement le mot, la syllabe et le cluster de caractères, comme unité de base de la modélisation.

Table 3 : Résultats en changeant les unités lexicales pour la modélisation statistique du langage

Modèle de langage	Modèle acoustique	CCER
LMmot	Grapheme_CD	20,8
LMsyl	Grapheme_CD	25,0
LMcc	Grapheme_CD	32,3
LMmot	Phoneme_CD	19,1
LMsyl	Phoneme_CD	26,3

Les résultats de la table 3 montrent que le mot reste la meilleure unité malgré les erreurs de segmentation. Une explication possible est qu'un mot khmer se compose en moyenne de 3,2 syllabes et de 4,3 clusters de caractères. Par conséquent, la couverture du modèle trigramme de syllabes et de clusters de caractères est beaucoup plus réduite que celle du modèle trigramme de mots. Le bénéfice de l'utilisation d'unités sous-lexicales pourrait cependant être observé quand le taux des mots hors vocabulaire est élevé (travaux en cours).

5.4. Modèles de langage hybrides

Dans cette expérimentation, des modèles hybrides sont créés en combinant les modèles de mots et de clusters de caractères. Les vocabulaires hybrides sont créés en ajoutant progressivement les N mots les plus fréquents du corpus de texte dans le vocabulaire de clusters de caractères V_0. En augmentant N de 0 à 20k, 6 différents

vocabulaires hybrides sont créés : $V_{0,}$ $V_{1k,}$ $V_{5k,}$ $V_{10k,}$ $V_{15k,}$ V_{20k}. 6 modèles de langage sont ensuite appris en utilisant ces vocabulaires hybrides avec le corpus de texte re-segmenté selon la version du vocabulaire utilisé. La figure 1 présente les résultats de test de ces 6 modèles de langage avec le modèle acoustique Grapheme_CD. On constate que la performance augmente au fur et à mesure que les mots sont introduits dans le vocabulaire. Les modèles hybrides V_{15k} et V_{20k} sont légèrement meilleurs que le modèle de mots mais la différence n'est pas significative. Il est aussi intéressant de noter que le modèle V5k, beaucoup plus petit, obtient quasiment les mêmes performances que le modèle de mots.

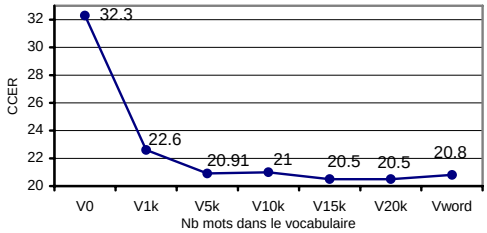

Figure 1 : Résultats avec des modèles hybrides

5.5. Combinaison des hypothèses

Afin d'étudier le potentiel de la combinaison des unités lexicales et sous-lexicales, nous appliquons une méthode qui combine des listes N-best et tente d'en extraire les meilleures hypothèses. D'abord, chaque système décode une liste de 20 meilleures hypothèses et toutes les hypothèses sont ramenées à l'unité la plus petite commune (le cluster de caractères ou la syllabe suivant les expérimentations). Nous combinons ensuite les sorties des deux systèmes. Avec une approche similaire à ROVER [14], nous appliquons un algorithme de vote simple basé sur le nombre d'occurrences des unités dans la liste N-best pour décoder la meilleure hypothèse. Le tableau 4 donne les CCER et le CCER Oracle des combinaisons des différents systèmes.

Table 4 : Résultats de la combinaison des systèmes

La combinaison des N-best listes	CCER	Oracle
20 best LMmot + 20 best LMsyl	21,2	12,1
20 best LMmot + 20 best LMcc	23,3	11,8
20 best LMsyl + 20 best LMcc	27,5	15,0
20 best LMmot+20 best LMsyl+20 best LMcc	23,5	10,8

Bien que le CCER Oracle montre le potentiel de cette approche de combinaisons, la méthode de vote simple ne permet pas d'améliorer la performance. Un mécanisme de combinaisons des systèmes plus évolué doit être étudié pour montrer l'avantage de cette approche. Une raison de cet échec est également dûe aux performances trop faibles de modèles LMsyl et LMcc par rapport au modèle de mots, ce qui rend la fusion inefficace. Nous travaillons actuellement à rehausser les performances des modèles LMsyl et LMcc par rescoring d'hypothèses en utilisant des modèles ngrammes d'ordre plus élevé que 3.

6. CONCLUSION

Cet article présente le développement d'un système de RAP pour le khmer. Une méthode de collecte rapide de données linguistiques a été décrite. Pour la modélisation acoustique, nous avons montré que la modélisation à base de graphèmes présente un bon potentiel dans le cas de cette langue peu dotée. Pour traiter le problème du manque de données et des erreurs de segmentation dans la modélisation du langage, nous avons essayé d'exploiter différentes vues sur les données en utilisant les unités lexicales et sous-lexicales. Les résultats des tests ont montré que le mot reste la meilleure unité de modélisation. Au niveau du système, bien que l'Oracle CCER montre le potentiel de l'approche de combinaisons, le vote simple ne permet pas d'obtenir une meilleure performance. Une solution de combinaisons de systèmes plus avancée qui fusionne les treillis est en cours d'élaboration.

BIBLIOGRAPHIE

[1] D. Vaufreydaz. Modélisation statistique du langage à partir d'Internet pour la reconnaissance automatique de la parole continue. *Thèse de doctorat de l'Université J. Fourier - Grenoble I*, France, 2002.

[2] X. Zhu and R. Rosenfeld. Improving Trigram Language Modelling with the World Wide Web. In *Proc. ICASSP*, pages 533-536, Salt Lake, USA, Mai 2001.

[3] www-clips.imag.fr/geod/User/brigitte.bigi/logiciel.html

[4] Barras and all. Transcriber: development and use of a tool for assisting speech corpora production. In *Speech Communication* Vol 33, No 1-2, 2000.

[5] M. Kurimo and all. Unsupervised segmentation of words into morphemes - Morpho Challenge 2005: Application to Automatic Speech Recognition. In *Proc. Interspeech*, pages 1021-1024, Pittsburgh, PA, 2006

[6] N. Abdillahi and all. Automatic transcription of Somali language. In *Proc. Interspeech*, pages 289-292, Pittsburgh, 2006.

[7] M. Afify and all. On the use of morphological analysis for dialectal Arabic Speech Recognition. In *Proc. Interspeech* pages 277-280, Pittsburgh, PA, 2006.

[8] E. Denoual and Y. Lepage. The character as an appropriate unit of processing for non-segmenting languages. *NLP Annual Meeting*, pages 731-734, Tokyo, Japan, 2006.

[9] Billa J. and all. Audio indexing of Arabic broadcast news. In *Proc. IEEE International Conference on Acoustique, Speech and Signal Processing*. Pages 5-8, Orlando, 2002

[10] M. Bisani and H. Ney. Multigram-based grapheme-to-phoneme conversion for LVCSR. In *Proc. EUROSPEECH*. Pages 933-936 Geneva, Switzerland, 2003

[11] Huffman, Franklin. *Cambodian System of Writing and Beginning Reader*. Yale University Press, 1970

[12] S. Seng and S. Sam. Traitement Automatique de la Langue Khmer. *Rapport de Projet AUF TALK 2ème tranche*". 2006

[13] http://cmusphinx.sourceforge.net/html/cmusphinx.php

[14] J.G Fiscus. A Post-Processing System to Yield Reduced Word Error Rates: Recogniser Output Voting Error Reduction (ROVER). *Proc. IEEE ASRU Workshop*, pages 347-352, USA, 1997