Un modèle syllabique pour la reconnaissance de l'écriture

Wassim Swaileh
Normandie Université, université de Rouen, LITIS EA 4108,
Campus du Madrillet,
76800 saint Étienne du Rouvray
wassim.swaileh2@univ-rouen.fr

Kamel Ait Mohand
Laboratoire Géography-cité CNRS - Paris
kamel.ait-mohand@parisgeo.cnrs.fr

Thierry Paquet
Normandie Université, université de Rouen, Laboratoire LITIS EA 4108,
Campus du Madrillet
76800 saint Étienne du Rouvray
thierry.paquet@univ-rouen.fr

RÉSUMÉ. *Dans cet article nous introduisons un nouveau modèle syllabique pour la reconnaissance de l'écriture. Une méthode de syllabation orthographique supervisée du Français est proposé pour la construction d'un vocabulaire de syllabes. Un modèle de langage statistique en n-gram combinant syllabes et caractères est appris sur un corpus Wikipedia. Le système de reconnaissance d'écriture fondé sur des modèles optiques HMM de caractères procède à un décodage en deux passes en exploitant le modèle syllabique proposé. L'évaluation est réalisée sur la base RIMES en analysant les performances pour différents taux de couverture du modèle syllabique. Nous comparons le modèle proposé à un modèle lexical ainsi qu'à un modèle de caractères. L'approche proposée permet d'atteindre des performances intéressantes grâce à sa capacité à couvrir une proportion importante des mots hors lexique en travaillant avec un lexique de syllabes de taille limitée combiné à un modèle de n-gram d'ordre raisonnable.*

ABSTRACT. *In this paper, we introduce a new syllabic model for handwriting recognition. We propose a supervised syllabification approach of the French language for building a vocabulary of syllables. A statistical n-gram language model of syllables is trained on a Wikipedia corpus. The handwriting recognition system, based on optical character HMM, performs a two pass decoding, integrating the proposed syllabic model. Evaluation is carried out on the RIMES dataset by analysing the performance for various coverage of the syllable model. We also compare the model with lexicon and character n-gram models. The proposed approach achieves interesting performance thanks to its capacity to cover a large amount of out of vocabulary words working with a limited amount of syllables combined with statistical n-gram of reasonable order.*

MOTS-CLÉS : *Syllabe, Syllabation, Ecriture manuscrite, Modèle de langage, Reconnaissance*

KEYWORDS: *Syllable, Syllabification, Handwriting, Language model, Recognition*

1. Introduction

La reconnaissance de l'écriture manuscrite fait l'objet de recherches depuis plusieurs dizaines d'années dans le but de transformer les images de textes manuscrits en leurs transcriptions numériques codées en ASCII ou UNICODE. Pour cela, l'idée générale des systèmes de reconnaissance d'écriture consiste à représenter les propriétés de l'image par des modèles probabilistes que l'on désigne souvent sous le terme de modèles optiques des caractères à reconnaître dans la langue considérée. A travers un processus d'apprentissage, les modèles optiques sont optimisés sur un ensemble d'exemples annotés afin de réaliser le mieux possible la tâche de transcription des images en leur représentation textuelle correspondante.

Dans la littérature, différentes structures ont été proposées pour réaliser des systèmes de reconnaissance d'écriture [Plötz et Fink 2009]. Selon leur structure lexicale, on peut classer les systèmes de la reconnaissance d'écriture en trois catégories principales. La première catégorie regroupe les systèmes à vocabulaire fermé qui sont optimisés pour faire la reconnaissance des mots dans un vocabulaire restreint et statique. Ce genre de système est mis en oeuvre pour des applications spécifiques comme la lecture des chèques bancaires [Gorski et al. 1999]. Dans ce cas les modèles optiques peuvent être des modèles de mots. On parle aussi de reconnaissance globale ou « holistique » pour ces approches.

La seconde catégorie regroupe les systèmes à vocabulaire dynamique qui sont capables de reconnaître des mots jamais vus par le système au moment de l'apprentissage. Pour cela les modèles optiques sont des modèles de caractères, et l'on parle de reconnaissance analytique pour désigner ces approches qui sont guidées par la connaissance d'un lexique (« lexicon driven ») au moment de la reconnaissance. Grâce à cette capacité, ces systèmes sont utilisables pour des applications généralistes, comme la reconnaissance de documents historiques par exemple [Pantke et al. 2013]. En fonction de l'accroissement de la taille du vocabulaire de l'application, la tâche de reconnaissance devient de plus en plus complexe, et elle voit ses performances diminuer, car des mots de plus en plus ressemblants doivent être discriminés par le système, et la compétition entre les mots devient de plus en plus difficile [Koerich, Sabourin, et Suen, 2003]. Pour pallier cette diminution de performances il est possible de contraindre le système de reconnaissance par un modèle de phrases qui modélise l'enchaînement des mots. On parle alors de modèle de langage, et la complexité du système de reconnaissance augmente encore, puisqu'il doit gérer en même temps les modèles optiques, le vocabulaire, et le modèle de langage [Plötz et Fink 2009, Kozielski et al. 2014].

La troisième catégorie d'approches regroupe les systèmes sans vocabulaire (« lexicon free») qui procèdent à la reconnaissance des mots dans une ligne de texte en reconnaissant l'enchaînement des caractères. Pour améliorer leurs performances ces systèmes peuvent recourir à des modèles de séquences de caractères sous la forme de modèles statistiques de n-gram [Plötz et Fink 2009] en considérant l'espace entre les mots comme un caractère [Brakensiek, Rottland, et Rigoll 2002].

L'avantage de ces systèmes est leur capacité à reconnaître n'importe quelle séquence de caractères, dont notamment les mots hors vocabulaire tels que les entités nommées, mais ils ont cependant l'inconvénient d'être moins performants que les modèles précédants en l'absence d'un niveau de modélisation de la phrase.

[Kozielski et al. 2014] ont exploré l'utilisation de modèles de langage de caractères (pour l'Anglais et l'Arabe) en utilisant des 10-gram de caractères estimés selon la méthode de Witten-Bell. Ils ont comparé cette approche sans lexique avec une approche avec lexique et un modèle en 3-gram estimé selon la méthode de Kneser-Ney modifiée. Ils ont également combiné les deux modèles (caractères et mots) en utilisant deux approches. La première en construisant un modèle global par interpolation des deux modèles, la seconde en utilisant une combinaison par modèles de repli (Back-off). Les résultats montrent l'efficacité de la combinaison des deux modèles de langages par interpolation. Souvent, pour diminuer l'effet néfaste des mots hors vocabulaire, on tente d'augmenter la taille du vocabulaire qui pilote le système de reconnaissance, mais cela est au détriment de la complexité des calculs et des risques de confusions associées [Rosenfeld 1995], tout en sachant que le vocabulaire n'est jamais complet. Une approche alternative visant à optimiser le compromis performance / taille du lexique, consiste à travailler avec des modèles de parties de mots. La reconnaissance des mots hors vocabulaire devient alors possible au niveau des parties de mots [Prasad et al. 2008] et elle peut être guidée par un modèle de langage spécifique modélisant les séquences possibles de parties de mots. Ce type d'approche n'est intéressant que pour des langues suffisamment fléchies, comme par exemple la langue Arabe. Plusieurs approches ont été proposées dans la littérature.

[Hamdani, Mousa, et Ney 2013] ont proposé un système de reconnaissance de l'écriture Arabe qui se base sur des modèles HMMs avec un modèle de langage des parties de mots arabes. Le vocabulaire utilisé contient des mots et des sous-parties de mots produits par une méthode de décomposition morphologique spécifique pour la langue Arabe. La décomposition se base sur la définition morphologique des racines, des suffixes et des préfixes des mots [Creutz et al. 2007]. Les résultats montrent l'amélioration apportée par ce système au niveau des mots hors lexique en comparaison d'un système de reconnaissance dirigé par un lexique de mots.

[BenZeghiba, Louradour, et Kermorvant 2015] ont proposé un modèle de langage Hybride pour l'Arabe qui est construit selon la fréquence observée des mots. L'idée est de garder les plus fréquents tel quel sans décomposition, et de décomposer uniquement les mots les moins fréquents en sous-parties de mots. En profitant de la propriété spécifique de la langue et de l'écriture Arabe, on défini un PAW (Part of Arabic Word) par une séquence de caractères pouvant être ligaturés entre eux. Un caractère ne pouvant être ligaturé avec un suivant définit la fin d'une partie de mot arabe [AbdulKader 2008]. L'avantage d'un modèle hybride mot / PAW tient au fait qu'on obtient un modèle de langage de bonne qualité tout en gardant une taille de vocabulaire réduite. Les deux modèles (hybride mot / PAW) et PAW seul obtiennent presque les mêmes performances sur les mots hors lexique mais le système hybride est moins complexe.

Dans cet article nous nous inspirons des travaux déjà réalisés sur la langue arabe pour proposer un modèle de reconnaissance de l'écriture manuscrite en langue Française fondé sur un modèle de partie de mots. Ce modèle repose sur la modélisation syllabique orthographique du Français qui s'appuie elle-même sur la modélisation phonétique de la langue. L'enjeu est de produire un modèle de langage pour un vocabulaire de syllabes de taille raisonnable qui soit capable de contraindre efficacement le système de reconnaissance optique pour lui conférer des performances intéressantes sur les mots hors vocabulaire. Pour construire le lexique de syllabes, nous proposons une méthode de syllabation orthographique supervisée exploitant la base de donnée lexicales informatisée Lexique3 [New et al., s. d.]. Trois configurations du système de reconnaissance sont évaluées ; caractères, syllabes et mots. Les tests sont effectués sur la RIMES 2011 [Grosicki et El-Abed 2011]. Les estimations des modèles de langage utilisés pour chacune des configurations du système de reconnaissance sont effectuées sur le lexique fermé de la base RIMES ainsi que sur différents lexiques ouverts constitués à partir de Wikipedia [« Wikimedia Downloads » 2015].

Le plan de cet article est le suivant: la base théorique du modèle syllabique est présentée au paragraphe 2, la méthode de syllabation proposée est décrite par la suite au paragraphe 3. Nous présentons la structure du système de reconnaissance dans le paragraphe 4. Les expérimentations sont présentées et analysées dans le paragraphe 5 avant de dresser différentes perspectives à ce travail.

2. Modélisation syllabique du Français

La syllabe jouerait un rôle important dans l'organisation de la parole et de la langue [Angoujard, 1997, d'après Ryst, 2014]. Le terme « syllabe » est parfois défini physiologiquement comme une unité ininterrompue du langage oral qui est constituée d'un son ou un groupe de sons prononcé en un seul souffle [BrownKeith 2006, Meynadier 2001]. La segmentation de la parole peut se faire en syllabes tant au niveau acoustique que phonologique [Ridouane, Meynadier, Fougeron, 2011], et les syllabes produites par ces deux modèles ne sont pas systématiquement compatibles [Ryst 2014]. La plupart des phonéticiens s'accordent sur le fait qu'une syllabe est composée fondamentalement d'une rime qui est précédée d'une attaque (une ou plusieurs consonnes « C » facultatives en début de syllabe). A l'intérieur d'une rime, le noyau (généralement une voyelle « V ») est l'élément constitutif de la syllabe. Il est suivi par une coda (une ou plusieurs consonnes « C » à la fin de la syllabe) [Ryst 2014]. Les langages diffèrent les uns des autres par rapport aux paramètres topologiques comme l'optionalité des attaques et la recevabilité des codas. Par exemple, les attaques sont obligatoires en Allemand alors que les codas sont interdites en Espagnol [Bartlett, Kondrak, et Cherry 2009]. En français, le noyau est toujours, semble-t-il, une voyelle. Ainsi, pour compter le nombre de syllabes dans un énoncé en français, il suffirait de compter le nombre de voyelles prononcées [Ryst 2014].

A l'écrit, des règles de césures typographiques (hyphenation en anglais) sont utilisées pour couper en deux un mot afin qu'il s'étende sur deux lignes de texte successives. La règle de césure impose de couper un mot entre deux syllabes orthographiques consécutives. Selon [Flipo, Bernard Gaulle, et Karine Vancauwenberghe 1994], la syllabe orthographique diffère de la syllabe phonétique parce qu'elle conserve tout « e » muet placé entre deux consonnes ou en fin de mot. La règle de césure sépare les consonnes doubles même si elles sont prononcées comme une consonne simple. Par exemple, on distingue graphiquement trois syllabes dans pu-re-té même si l'on prononce [pyr-te] (deux syllabes phonétiques). [Roekhaut et Richard Beaufort 2012] ont classé les syllabes en trois catégories différentes :

1. Une ***syllabe phonétique*** qui est composée d'un regroupement de phonèmes qui se prononcent en une seule émission.

2. Une ***syllabe graphémique*** qui représente une transposition fidèle de la syllabation phonétique dans l'orthographe du mot

3. Une ***syllabe orthographique*** qui applique les règles de césure qui doivent être respectées à l'écrit.

Il semble difficile de concilier ces différents points de vue de spécialistes, mais en tout état de cause, seules les descriptions en syllabes graphémiques ou en syllabes orthographiques proposent une décomposition de l'écrit susceptible d'impacter un système de reconnaissance.

Dans cette étude, nous avons choisi d'utiliser la représentation orthographique syllabée de la base lexicale informatisée Lexique3 [New et al., s. d.]. Cette base propose une décomposition syllabique orthographique pour un lexique de près de 142 695 mots Français. Elle constitue donc une base de connaissance irremplaçable à partir de laquelle notre modèle syllabique est élaboré. Cependant, en dépit de sa taille relativement importante, on constate rapidement que cette base ne couvre pas le vocabulaire de la langue Française. Dans le cas qui nous intéresse plus particulièrement, cette base ne couvre pas le vocabulaire de la base RIMES, qui constitue l'un des ensembles d'apprentissage et de test de référence pour la reconnaissance de l'écriture manuscrites.

D'une manière générale il nous faut trouver un moyen de générer un modèle syllabique pouvant couvrir totalement, ou de manière paramétrable, un corpus quelconque. Pour cela il est nécessaire d'élaborer une méthode automatique de syllabation du Français. Celle-ci pourra alors être mise en œuvre sur des lexiques quelconques pour proposer un lexique de syllabes approprié.

La méthode que nous proposons est une méthode supervisée qui exploite la base Lexique3 et une mesure de similarité entre les mots.

3. Une méthode de syllabation automatique supervisée

La méthode de syllabation automatique supervisée que nous proposons se base sur la recherche des structures lexicales et phonétiques similaires pour proposer une segmentation en syllabes d'un mot inconnu. Nous disposons d'un lexique de mots $L = \{(m_1, s_1), (m_2, s_2), \dots, (m_m, s_m), (m_l, s_l),\}$ représentés par leur séquence de caractères m_n , associés à leur décomposition syllabique, représentée par leur séquence de syllabes s_n. Nous souhaitons déterminer la séquence de syllabes s d'un mot m ne figurant pas dans le lexique L.

Une première idée est de rechercher le mot m_n du dictionnaire le plus proche du mot inconnu. Mais deux mots très similaires peuvent avoir des décompositions syllabiques différentes notamment s'ils diffèrent d'une voyelle, qui marque souvent la présence d'une syllabe. Pour prendre en compte cette information nous introduisons un structure orthographique représentant le mot par sa séquence en consonnes et voyelles. Par exemple, le mot $m =$ « *Bonjour* » est codé par sa structure orthographique $ss =$ « *CVCCVVC* ». On construit alors une mesure de similarité combinant les deux représentations selon la formule suivante, où S_{lex} et S_{syl} sont deux mesures de similarité sur les représentations lexicales et syllabiques :

$$S_G\big((m, ss), (m_i, ss_i)\big) = \frac{S_{lex}(m, m_i) + S_{syl}(ss, ss_i)}{2} \qquad (1)$$

Le score de similarité entre séquences de caractères comptabilise le nombre moyen de couples de caractères identiques, aux mêmes positions, entre les deux séquences. Lorsque les séquences sont de tailles différentes, la plus courte est complétée à la fin par des caractères vides, pour que les deux séquences soient de la même taille. De cette façon on réalise le découpage en syllabes en se basant sur le préfixe du mot du dictionnaire, les erreurs de découpage en syllabes sont possibles sur les suffixes qui peuvent être différents du fait de la complétion avec des espaces en fin de mots.

Lorsque l'entrée du lexique la plus proche du mot inconnu obtient un score de similarité supérieur à un seuil T, sa représentation syllabique sert de modèle pour décomposer le mot inconnu. Plus exactement, la segmentation en syllabes du mot inconnu est réalisée aux mêmes positions que dans le mot issu du lexique. Lorsque le score de similarité est inférieur au seuil T, la décomposition en caractères est admise comme décomposition syllabique par défaut.

Le tableau 1 ci-dessous, donne quelques exemples de résultats obtenus par la méthode proposée sur des mots non syllabés de la base Lexiques3. Certains mots de cette base n'ont en effet pas de description syllabique proposée par l'algorithme de syllabation utilisé sur la forme phonologique [New et al., s. d.]. Ces résultats semblent tout à fait cohérents, bien que la validation de ces résultats par un expert reste à faire.

Mot requête	Candidat de lexique3, et sa syllabation	Syllabation proposée
Bonjour	*Toujours (Tou-jours)*	*Bon-jour*
conceptuellement	*Conventuellement (con-ven-tuel-le-ment)*	*con-cep-tuel-le-ment*
dérogatoire	*dédicatoire (dé-di-ca-toi-re)*	*dé-ro-ga-toi-re*
encéphalopathie	*Encéphalogramme (en-cé-pha-lo-gram-me)*	*en-cé-pha-lo-path-ie*

Tableau 1: Exemples de mots syllabés par la méthode proposée, et non renseignés dans la base Lexique3.

4. Le système de reconnaissance d'écriture

Notre système de reconnaissance se base sur une modélisation optique des caractères fondée sur les modèles statistiques de Markov cachés (Hidden Markov Model, ou HMM en anglais). Les composantes essentielles dans la construction de notre système sont les caractères alphanumériques. Nous avons au total 100 modèles de caractères différents lors des expérimentations sur la base RIMES, en considérant l'espace entre les mots comme un caractère.

Notre système de reconnaissance est construit en quatre étapes principales ; les prétraitements, l'apprentissage des modèles optiques, la génération du vocabulaire et l'apprentissage du modèle de langage. L'étape de reconnaissance est réalisée selon un algorithme de décodage en deux passes.

4.1 *Pré-traitements*

Lors des pré-traitements nous procédons à la localisation des lignes dans les blocs de textes afin d'améliorer l'indexation rectangulaire fournie dans la base RIMES, car celle-ci fournit des lignes assez bruitées.

En effet, si on se limite à extraire les zones rectangulaires on obtient des lignes contenant des chevauchements avec les lignes précédentes ou suivantes à l'endroit des ascendants ou descendants. La méthode de segmentation automatique en lignes est décrite dans [Swaileh, Mohand, et Paquet 2015]. Ensuite, toutes les images de lignes sont redressées horizontalement et verticalement (*deskew* et *deslant*) puis normalisées à une hauteur de 96 pixels.

4.2 *Modèles optiques de caractères*

Les modèles optiques exploitent les caractéristiques HoG (Histogram of Gradients) extraites à l'aide d'une fenêtre glissante de 20 pixels de largeur. Le décalage entre deux positions successives est de 2 pixels. Chaque trame est décrite par un vecteur de 70 caractéristiques réelles. 64 caractéristiques représentent la description HoG, et 6 caractéristiques codent une description géométrique de la trame.

Généralement, la structure interne des modèles optiques (HMM) est caractérisée par un nombre fixe d'états cachés et pour chacun, un mélange de Gaussiennes de taille fixe également. Nous avons choisi d'utiliser des mélanges de 20 Gaussiennes, qui garantissent un pouvoir de description assez précis de chaque trame.

La détermination du nombre d'états cachés est un problème d'optimisation. Un nombre d'états surestimé conduit à un sur-apprentissage des modèles. Un nombre d'états sous-estimé conduit à des modèles insuffisamment spécialisés. Ce problème a été abordé dans [Zimmermann et Bunke 2002], [Cirera, Fornes, et Llados 2015], [Ait-Mohand, Paquet, et Ragot 2014]. Nous nous inspirons ici de la méthode proposée dans la première référence qui se fonde sur la méthode de Bakis pour optimiser le nombre d'états de chaque modèle de caractère.

Nous calculons le nombre moyen T de trames par chaque modèle optique lors d'un alignement forcé du modèle correspondant à la vérité terrain de chaque image sur la séquence de trames. Le nombre d'états E du modèle correspondant est ensuite défini comme une fraction de T ($E = \alpha.T$).

Un nouvel apprentissage (Baum-Welch) des nouveaux modèles ainsi paramétrés selon α est effectué. Puis on procède enfin à un décodage sans lexique des données avec les modèles appris et on déduit le taux de reconnaissance des caractères. L'opération est répétée pour différentes valeurs de α (par valeurs croissantes entre 0 et 1) et on sélectionne finalement les modèles les plus performants en fonction d'un critère combinant le taux moyen de reconnaissance de caractères et le taux d'alignement des modèles sur les exemples d'apprentissage. En effet, des modèles trop longs ont tendance à maximiser le taux de reconnaissance mais à générer des défauts d'alignement sur les exemples les plus courts. Ce critère est testé à chaque itération de l'apprentissage Baum-Welch. L'apprentissage est stoppé au moment où le critère passe par son maximum. On obtient alors les modèles optiques optimisés.

L'apprentissage des modèles optiques est réalisé sur la base d'apprentissage RIMES 2011 qui contient 10963 images de lignes de texte étiquetées qui sont segmentées à partir de 1500 images de paragraphes écrits par différents scripteurs dans différentes conditions d'écriture.

4.3 *Lexiques et modèles de langage*

La troisième étape de construction de notre système concerne l'établissement des vocabulaires et des modèles de langage qui seront utilisés par le système durant le décodage. Deux corpus de textes sont utilisés pour la génération des vocabulaires et des modèles de langage. Un premier corpus rassemble les textes de la base RIMES (base d'apprentissage et base de validation). Un second corpus, beaucoup plus important en taille, rassemble des textes collectés de la base Wikipédia. A partir de ces deux corpus, nous avons généré trois configurations de vocabulaire et de modèle de langage.

La première configuration est une configuration sans lexique, qui modélise les séquences de caractères à l'aide d'un modèle de n-gram de caractères. Les modèles n-gram sont estimés sur les corpus RIMES ou Wikipédia. La seconde configuration est une configuration avec lexique de mots.

Les vocabulaires sont des lexiques de mots de taille variable et les modèles de langages sont des n-grams de mots estimés sur les corpus RIMES et Wikipédia. La troisième configuration est une configuration procédant selon le modèle syllabique que nous proposons. Les vocabulaires sont des lexiques de syllabes obtenus à l'aide de la méthode de syllabation proposée ci-dessus, et les modèles de langage sont des n-grams de syllabes estimés sur les même corpus RIMES et Wikipédia.

4.4 *Etape de reconnaissance*

Notre système est caractérisé par un décodage en deux passes. La première passe traite l'exemple de test en procédant à un décodage selon l'algorithme de Viterbi avec élagage au cours du temps (*time synchronous Beam search Viterbi*).

Les modèles optiques sont utilisés seuls pour le modèle dit sans lexique, ou bien ils sont concaténés pour former les mots ou les syllabes du lexique de travail. Selon le modèle utilisé, l'algorithme de décodage tient compte d'un modèle bi-gram de caractères, de syllabes ou de mots, pour produire un réseau d'hypothèses de séquences de caractères, de syllabes ou de mots.

Deux paramètres essentiels guident cette première passe de décodage : le paramètre de mise à l'échelle du modèle de langage γ vis à vis du modèle optique, et le paramètre de pénalisation β en fin de mot qui contrôle l'insertion trop fréquente de mots courts. Ces deux paramètres doivent être optimisés pour un couplage optimal du modèle optique avec le modèle de langage considéré, car ces deux modèles sont estimés indépendamment l'un de l'autre lors de l'apprentissage.

La seconde passe de décodage analyse le réseau d'hypothèses fourni par la première passe en utilisant un modèle de langage en n-gram d'ordre plus élevé qui permet de re-pondérer les premières hypothèses. Cette dernière étape fournit la solution finale de reconnaissance de la ligne de texte.

Lors du décodage on cherche la séquence de mots $\widehat{W}$ qui maximise la probabilité *a posteriori* $P(W|S)$ parmi toutes les phrases possibles W. En utilisant la formule de Bayes et en introduisant les deux hyper-paramètres définis précédemmant, on arrive finalement à la formule ci-dessous qui régit l'étape de décodage. Dans cette formule, S represente la séquence d'observations extraites de l'image et $P(S|W)$représente la vraisemblance que les caracteristiques S soient générées par la phrase W, elle est déduite du modèle optique. $P(W)$ est la probabilité *a priori* de la phrase W, elle est déduite du modèle de langage.

$$\widehat{W} = argmax_w \, P(S|W) \, P(W)^{\gamma} \, \beta \qquad (2)$$

5. Evaluation

Pour optimiser et tester les performances de notre système, nous avons utilisé la base de validation de la base RIMES qui contient 764 lignes extraites de 100 images de paragraphes. La moitié de cette base de validation a été utilisée pour l'optimisation des paramètres de décodage et l'autre moitié a été utilisée pour calculer les performances des différents systèmes.

Nous avons défini une première configuration de nos systèmes en combinant les ressources de la base RIMES et de la base Wikipédia (Lexiques et modèles de langage). Elle permettra une évaluation des trois modèles concurrents (caractères, syllabes et mots) en condition de lexique et de modèle de langage fermés mais pour différentes taille de lexiques, en ajoutant au lexique de la base RIMES différents ensembles constitués des mots les plus fréquents de la base Wikipédia (10K, 20K, 40K et 60K mots les plus fréquents). Pour chaque taille de lexique, un modèle de langage est entraîné spécifiquement sur les corpus Wikipédia et RIMES.

La seconde configuration de notre système permet d'évaluer les performances des différents modèles dans des situations où le lexique de la base de test est partiellement couvert par le lexique de travail du système. Dans ce cas on utilise uniquement les ressources wikipedia pour déterminer des lexiques et leurs modèles de langage associés. Nous avons retenu les lexiques wikipedia contenant les mots les plus fréquents (10K, 20K, 40K et 60K mots).

Le tableau 2 donne le taux de couverture du lexique RIMES et le taux de couverture de la base RIMES qui tient compte de la fréquence des mots. On voit que la base RIMES présente un taux de mots hors lexique assez important, même pour des lexiques de grande taille. Ceci est du à la présence de nombreuses entités nommées, nom de personnes, nom de villes, dates, séquences alphanumériques.

Corpus Wikipedia	10K mots	20K mots	40K mots	60K mots	296K mots
Taux de couverture du lexique RIMES (en mots)	43.92 %	55.64 %	64.71 %	69.25 %	79.51 %
Taux de couverture de la base RIME (en mots)	86.52 %	89.96 %	92.92 %	97.1 %	97.78 %

Tableau 2: Taux de couverture du lexique et de la base RIMES par les lexiques Wikipédia.

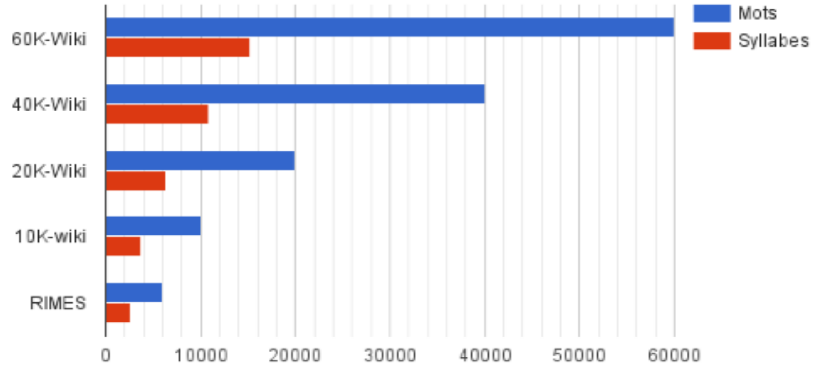

Figure 1: Taille des lexiques de syllabes dérivés des lexiques mots Wikipédia.

Le figure 1 donne la taille des lexiques de syllabes dérivés des lexiques de mots. On observe immédiatement la réduction des tailles des lexiques (de l'ordre de deux tiers) en travaillant sur un modèle syllabique.

Le taux d'erreur mot (word error rate, WER) est utilisé pour évaluer la performance de nos systèmes de reconnaissance. Figure 2 illustre le comportement d système pour chaque modèle (caractères, syllabes et mots) et pour différentes tailles de lexiques fermés. On observe pour la configuration de lexique la plus spécialisée à la base RIMES, que le modèle syllabique obtient des performances (WER=20.6%) légèrement inférieures au modèle mot (WER=19.6%), et qu'il est nettement meilleur que le modèle caractère (WER=27.8%).

Cette tendance se confirme lorsque l'on augmente la taille du vocabulaire. On observe même que les performances du modèle syllabique ne se dégradent pas lorsque la taille du lexique augmente, contrairement au modèle mot. Ces performances sont obtenues pour des modèles n-gram d'ordre 6 pour les caractères, 6 pour les syllabes, et 3 pour les mots. On peut donc conclure qu'en vocabulaire fermé, le modèle syllabique est une très bonne alternative à un modèle lexical, puisqu'il obtient des performances voisines pour une complexité réduite (lexique réduit, et modèle de langage réduit en conséquence).

Nous cherchons maintenant à évaluer la qualité du modèle syllabique lorsqu'on travaille avec des vocabulaires qui couvrent partiellement les textes à transcrire, comme c'est le cas dans les situations réelles, où l'on ne peut notamment pas couvrir un certain nombre d'entités nommées.

La figure 3 donne le WER des trois modèles en utilisant les lexiques de la base Wikipédia et des modèles de langage optimisés sur le corpus Wikipédia uniquement. A titre indicatif les meilleures performances avec le lexique fermé RIMES sont rappelées. On observe que le modèle syllabique obtient des performances égales ou supérieures au modèle mot.

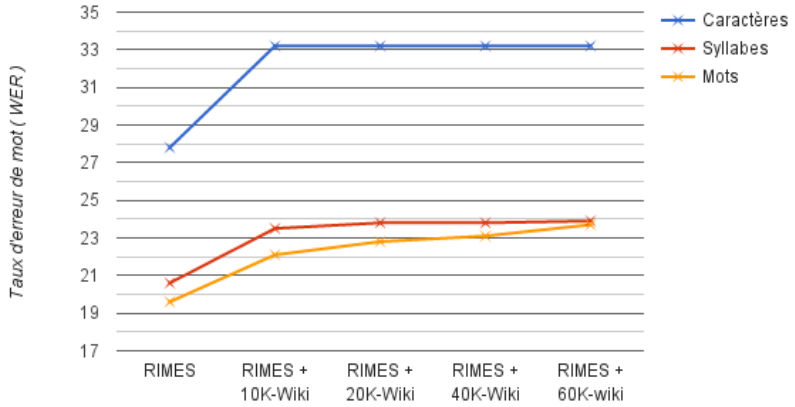

Figure 2: WER (%) des trois modèles, pour différentes tailles de lexiques fermés.

Les modèles sont équivalents lorsqu'on travaille avec un lexique de 60K, dans les autres situations, le modèle syllabique obtient de meilleures performances que le modèle mots car il permet de couvrir des mots hors lexique, ce que le modèle mots ne peut pas faire.

Une fois encore on remarque que le modèle syllabique obtient des performances très stables quelle que soit la taille du lexique à partir duquel il est construit.

On peut donc conclure, comme nous cherchions à le démontrer, que le modèle syllabique offre une capacité de couverture lexicale très intéressante, notamment des mots hors lexique tout en étant de complexité inférieure à un modèle lexicale.

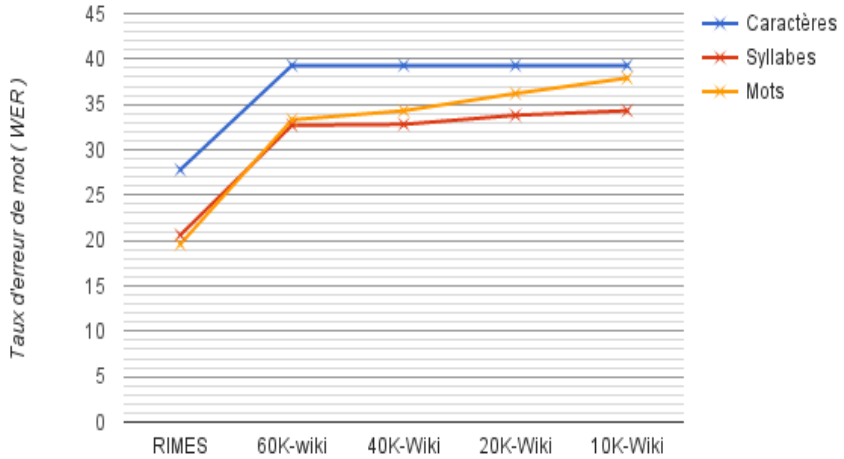

Figure 3: WER (%) des trois modèles, pour différentes tailles de lexiques couvrant partiellement la base RIMES (voir tableau 2).

6. Conclusion et perspectives

Dans cette étude nous avons proposé un modèle syllabique du Français pour la reconnaissance de l'écriture manuscrite. Ce modèle offre beaucoup d'avantages par rapport à un modèle de caractères qui modélise mal les mots, et par rapport à un modèle lexical qui ne modélise que les mots connus du lexique et du corpus d'apprentissage. Les avantages de ce modèle sont doubles. D'une part, il est de complexité limitée, puisqu'il travaille avec un lexique de syllabes réduit. Il en découle un modèle n-gram de syllabes lui-même plus compact, donc mieux paramétré, et donc plus facile à optimiser. D'autre part il offre des performances supérieures à un modèle lexical lorsqu'on travaille avec des mots hors lexique.

Pour générer le modèle syllabique nous nous sommes appuyé sur la base lexique3, qui repose sur une modélisation syllabique orthographique du Français. D'autres modèles pourraient être évalués à titre de comparaison. La question de la recherche d'un découpage optimal en parties de mots pour la tâche qui nous intéresse pourrait être explorée également. Par ailleurs, il conviendrait de savoir également si une telle modélisation offre le même intérêt pour d'autres langues. L'intérêt pour la langue Arabe ayant déjà été démontré comme indiqué dans l'étude bibliographique.

Remerciements

Nous Adressons nos remerciements à Mme. Elise Ryste et M. Christophe Coupeur qui nous ont offert leurs conseils en tant que spécialistes linguistes.

7. Bibliographie

AbdulKader, Ahmad. 2008. *A Two-Tier Arabic Offline Handwriting Recognition Based on Conditional Joining Rules*. Springer Berlin Heidelberg.

Angoujard, J.-P. (1997). *Théorie de la syllabe : rythme et qualité*. Collection Sciences du Langage, Paris : CNRS.

BenZeghiba, Mohamed Faouzi, Jerome Louradour, et Christopher Kermorvant. 2015. « *Hybrid word/Part-of-Arabic-Word Language Models for arabic text document recognition* ». In , 671-75. IEEE. doi:10.1109/ICDAR.2015.7333846.

Brakensiek, A., J. Rottland, et G. Rigoll. 2002. « *Handwritten address recognition with open vocabulary using character n-grams* ». In , 357-62. IEEE Comput. Soc.

BrownKeith. 2006. *Encyclopedia of Language and Linguistics*. Set Set. San Diego, Saint Louis, Elsevier Science & Technology Books; Elsevier Distributor.

Creutz, Mathias, Teemu Hirsimäki, Mikko Kurimo, Antti Puurula, Janne Pylkkönen, Vesa Siivola, Matti Varjokallio, Ebru Arisoy, Murat Saraçlar, et Andreas Stolcke. 2007. « *Morph-based speech recognition and modeling of out-of-vocabulary words across languages* ». ACM Transactions on Speech and Language Processing (TSLP) 5 (1): 3.

Gorski, N., V. Anisimov, E. Augustin, O. Baret, D. Price, et J.-C. Simon. 1999. « *A2iA Check Reader: A Family of Bank Check Recognition Systems* ». In , 523-26. IEEE.

Grosicki, Emmanuele, et Haikal El-Abed. 2011. « *ICDAR 2011 - French Handwriting Recognition Competition* ». In , 1459-63. IEEE. doi:10.1109/ICDAR.2011.290.

Hamdani, Mahdi, Amr El-Desoky Mousa, et Hermann Ney. 2013. « *Open Vocabulary Arabic Handwriting Recognition Using Morphological Decomposition* ». In , 280-84. IEEE.

Koerich, A. L., R. Sabourin, et C. Y. Suen. s. d. « Large Vocabulary off-Line Handwriting Recognition: A Survey ». Pattern Analysis & Applications 6 (2): 97-121.

Kozielski, Michal, Martin Matysiak, Patrick Doetsch, Ralf Schloter, et Hermann Ney. 2014. « *Open-Lexicon Language Modeling Combining Word and Character Levels* ». In , 343-48.

New, Boris, Christophe Pallier, Marc Brysbaert, et Ludovic Ferrand. s. d. « *Lexique 3: A New French Lexical Database* ». Behavior Research Methods, Instruments, & Computers

Pantke, Werner, Volker Märgner, Daniel Fecker, Tim Fingscheidt, Abedelkadir Asi, Ofer Biller, Jihad El-Sana, Raid Saabni, et Mohammad Yehia. 2013. « *HADARA - A Software System for Semi-Automatic Processing of Historical Handwritten Arabic Documents* ». Archiving 2013 Final Program and Proceedings.

Prasad, Rohit, Shirin Saleem, Matin Kamali, Ralf Meermeier, et Prem Natarajan. 2008. « *Improvements in Hidden Markov Model Based Arabic OCR* ». In , 1-4. IEEE.

Ridouane, R., Meynadier, Y. & Fougeron, C. (2011). *La syllabe : objet théorique et réalité physique*. Faits de langues, 37, La parole : origines, développement et structures, Paris : Ophrys, 213-234.

Rosenfeld, Roni. 1995. « *Optimizing Lexical and Ngram Coverage via Judicious Use of Linguistic Data* », Computer Science Department

« *Wikimedia Downloads* ». 2015. https://dumps.wikimedia.org/frwiki/latest/frwiki-latest-pages-articles.xml.bz2

Bartlett, Susan, Grzegorz Kondrak, et Colin Cherry. 2009. « *On the syllabification of phonemes* ». In , 308-16. Association for Computational Linguistics.

Flipo, Daniel, Bernard Gaulle, et Karine Vancauwenberghe. 1994. « *Motifs fran cais de c esure typographique* ». *Cahiers gut enberg n°18*.

Kozielski, Michal, Martin Matysiak, Patrick Doetsch, Ralf Schloter, et Hermann Ney. 2014. « *Open-Lexicon Language Modeling Combining Word and Character Levels* ». In , 343-48. IEEE.

Plötz, Thomas, et Gernot A. Fink. 2009. « *Markov models for offline handwriting recognition: a survey* ». International Journal on Document Analysis and Recognition (IJDAR), no Volume 12, Issue 4 (octobre).

Roekhaut, Sandrine Brognaux Sophie, et Richard Beaufort. 2012. « *Syllabation graphémique automatique à l'aide d'un dictionnaire phonétique aligné* ». Université catholique de Louvain – CENTAL – Groupe Norme

Ryst, 2014. *La syllabation en anglais et en français : considérations formelles et expérimentales*. Thèse de doctorat. Université Paris 8.

Ait-Mohand, Kamel, T. Paquet, et N. Ragot. 2014. « *Combining Structure and Parameter Adaptation of HMMs for Printed Text Recognition* ». Pattern Analysis and Machine Intelligence, IEEE Transactions on 36 (9): 1716-32.

Cirera, Nuria, Alicia Fornes, et Josep Llados. 2015. « *Hidden Markov model topology optimization for handwriting recognition* ». In , 626-30. IEEE.

Meynadier, Yohann. 2001. « *La syllabe phonétique et phonologique : une introduction* ». Travaux Interdisciplinaires du Laboratoire Parole et Langage d'Aix-en-Provence (TIPA) 20: 91-148.

Swaileh, Wassim, Kamel Ait Mohand, et Thierry Paquet. 2015. « *Multi-script iterative steerable directional filtering for handwritten text line extraction* ». In , 1241-45. IEEE.

Zimmermann, M., et H. Bunke. 2002. « *Hidden Markov model length optimization for handwriting recognition systems* ». In , 369-74. IEEE Comput. Soc.