Un modèle syllabique du français et de l'anglais pour la reconnaissance de l'écriture

Wassim Swaileh, Thierry Paquet

Dans Document numérique 2016/2-3 Vol. 19 , pages 117 à 134
Éditions JLE

ISSN 1279-5127
ISBN 9782746247956
Date de mise en ligne : 10/01/2017

Un modèle syllabique du français et de l'anglais pour la reconnaissance de l'écriture

Wassim Swaileh, Thierry Paquet

Normandie Université, Université de Rouen, LITIS EA 4108
Campus du Madrillet, 76800 Saint Étienne du Rouvray, France
wassim.swaileh2@univ-rouen.fr, thierry.paquet@univ-rouen.fr

RÉSUMÉ. Dans cet article nous introduisons une nouvelle méthode de modélisation du texte pour la reconnaissance de l'écriture. Une méthode de syllabation orthographique supervisée est proposée 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 Wikipédia. Le système de reconnaissance d'écriture fondé sur des modèles optiques HMM de caractères procède alors à un décodage en deux passes en exploitant le modèle syllabique proposé. L'évaluation est réalisée pour le français et l'anglais, sur les bases RIMES et IAM respectivement, en analysant les performances pour différents taux de couverture des modèles syllabiques. 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 modeling method of texts for handwriting recognition. We propose a supervised syllabification approach 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 HMM character models, performs a two pass decoding, integrating the proposed syllabic model. Evaluation is carried out for French and English using the RIMES and IAM datasets respectively, and by analysing the performance for various coverages 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 while working with a limited amount of syllables combined with statistical n-gram of reasonable order.

MOTS-CLÉS : syllabe, syllabation, reconnaissance de l'écriture manuscrite, modèle de langage.

KEYWORDS: syllable, syllabification, handwriting recognition, language model.

DOI:10.3166/DN.19.2-3.117-134

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 qui sont les modèles optiques des caractères à reconnaître dans la langue considérée. À 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 al*., 2009). Selon leur structure lexicale, on peut classer les systèmes de 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 œuvre 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 des 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 concurrence entre les mots devient de plus en plus difficile (Koerich *et al*., 2014). 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 al*., 2009)..

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 al*., 2009) en considérant l'espace entre les mots comme un caractère (Brakensiek *et al*., 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 précédents en l'absence d'un niveau de modélisation de la phrase.

Kozielski et son équipe (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 *et* Roni., 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 flexionnelles, comme par exemple la langue arabe. Plusieurs approches ont été proposées dans la littérature.

Hamdani (Hamdani *et al.*, 2013) a proposé un système de reconnaissance de l'écriture arabe qui se base sur des modèles HMM 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 (BenZeghiba *et al.*, 2015) a 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 tels quels 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 définit 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 *et* Ahmad, 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.

Feild (Feild *et al.*, 2013) a proposé un modèle syllabique probabiliste pour la reconnaissance de textes dans des images de scènes naturelles de différentes natures, comme par exemple des panneaux publicitaires. Ce modèle repose sur une grammaire probabiliste hors contexte pour modéliser les syllabes (voyelles et consonnes) et les mots anglais à partir des syllabes. Cette approche obtient de bonnes performances pour les mots hors lexique tels que des noms propres, à condition toutefois qu'ils soient de consonance anglaise. La grammaire est construite grâce à la définition de règles d'agencement des caractères pour former des syllabes. Les syllabes sont définies dans un dictionnaire. Le défaut principal de cette méthode est qu'elle ne permet pas de modéliser les enchaînements de syllabes dans les mots. Testé sur deux jeux de données différents constitués de mots isolés, le modèle améliore de 4 % les performances d'un système traditionnel reposant sur une reconnaissance dirigée par un lexique de mots.

Dans cet article nous nous inspirons des travaux déjà réalisés sur la langue arabe pour proposer un modèle de reconnaissance de l'écriture manuscrite fondé sur un modèle de parties de mots. Ce modèle se distingue cependant de ces travaux car il repose sur une modélisation syllabique orthographique 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 un dictionnaire de syllabes. Les expériences sont réalisées pour le français et pour l'anglais en utilisant respectivement les bases *Lexique3* (New *et al*., 2004) et *English Language Hyphenation Dictionary* (ELHD) (Hindson, 2006), pour la construction des modèles syllabiques. En ce qui concerne les tests de reconnaissance d'écriture, nous utilisons la base française RIMES (Grosicki *et al*., 2011) et la base anglaise IAM (Marti *et* Bunke, 2002). Trois configurations du système de reconnaissance sont évaluées en utilisant respectivement des modèles de langages de caractères, de syllabes et de mots. Les estimations des modèles de langage utilisés pour chacune des configurations sont réalisées sur les lexiques fermés ainsi que sur différents lexiques ouverts constitués à partir de Wikipédia.

Le plan de cet article est le suivant : la base théorique du modèle syllabique est présentée en section 2, la méthode de syllabation proposée est décrite section 3. Nous présentons la structure du système de reconnaissance en section 4. Les expérimentations sont présentées et analysées en section 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). 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 *et* Yohann, 2001). La segmentation de la parole peut se faire en syllabes tant au niveau

acoustique que phonologique (Ridouane *et al*., 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). À 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). 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 *et al*., 2009). En français, le noyau est toujours une voyelle semble-t-il. Ainsi, pour compter le nombre de syllabes dans un énoncé en français, il suffirait de compter le nombre de voyelles prononcées (Ryst, 2014).

À 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 (Flipo *et al.,* 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 (Roekhaut *et al*., 2012) a classé les syllabes en trois catégories différentes :

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

– Une *syllabe graphémique* qui représente une transposition fidèle de la syllabation phonétique dans le système d'écriture utilisé pour écrire le mot.

– 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.,* 2004) pour le français et le dictionnaire *English Language Hyphenation Dictionary,* ELHD, (Hindson, 2006) pour l'anglais. Lexique3 contient la décomposition syllabique orthographique de près de 146 950 mots, en 9 522 syllabes seulement. Le dictionnaire ELHD contient quant à lui 166 280 mots qui sont décomposés en 21 991 syllabes. Cependant, malgré leur taille relativement importante ces lexiques ne couvrent pas le français ni l'anglais. Par exemple, Lexique3 couvre seulement 69,83 % du vocabulaire de la base RIMES. De même ELHD couvre seulement 54,42 % du vocabulaire de la base IAM. Les syllables

permettent dans les deux cas d'atteindre des taux de couverture nettement plus élevés mais encore insuffisants : 80 % pour les mots de la base RIMES et 79 % pour les mots de la base IAM. D'une manière générale il nous faut donc 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. 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 un lexique syllabisé et une mesure de similarité entre les mots. Nous la présentons maintenant.

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 $L = \{(m_1, s_1), (m_2, s_2), \ldots, (m_m, s_m), \ldots, (m_l, s_l)\}$ 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 nous basons également sur la structure orthographique phonétique représentant le mot. Par exemple, le mot « **Bonjour** » est codé par sa structure orthographique phonétique ss = « **CVCCVVC** » dans laquelle les caractères sont simplement remplacés par leur catégorie phonétique (C pour consonne et V pour voyelle). On construit alors une mesure de similarité combinant les deux représentations selon la formule suivante, où S_{lex} et S_{syl} sont respectivement deux mesures de similarité sur les représentations lexicales et phonétiques :

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

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 mot.

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 choix du seuil T est détaillé dans la partie évaluation.

Dans le tableau 1 nous donnons quelques exemples de syllabations pour des mots français et anglais. Nous pouvons remarquer que notre méthode propose des syllabations correctes pour ces mots, mais une évaluation plus rigoureuse à l'aide d'un expert serait nécessaire sur un ensemble d'exemples plus important pour caractériser plus finement les propriétés de notre méthode.

Tableau 1. Exemples de mots syllabés par la méthode proposée

	Mot requête	Candidat de lexique3, et sa syllabation	Syllabation proposée
FR	Bonjour	Toujours → (Tou-jours)	Bon-jour
	Dérogatoire	Dédicatoire → (dé-di-ca-toi-re)	dé-ro-ga-toi-re
EN	Moved	Moped → (mo-ped)	mo-ved
	Answering	anglewing → (an-gle-wing)	An-swe-ring

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, et 80 modèles différents pour les expérimentations sur la base IAM, 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. Prétraitements

Lors des prétraitements 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 fournies dans les fichiers vérité terrain, 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 utilisée est décrite dans (Swaileh *et al*. 2015). Ensuite, toutes les images de lignes sont redressées horizontalement et verticalement (deskew et deslant) puis normalisées à une hauteur de 96 pixels. Pour la base IAM, la délimitation des lignes d'écritures par les rectangles englobants est correcte. Les images des lignes sont ensuite corrigées comme pour la base RIMES.

4.2. Modèles optiques de caractères

Les modèles optiques exploitent les caractéristiques HoG (*histogram of gradient*s) 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 surapprentissage 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 *et al*., 2015 ; Ait-Mohand *et al*., 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 alors réalisé. 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 les bases RIMES 2011 et IAM respectivement qui contiennent 10 963 et 13 353 images de lignes de texte étiquetées qui sont segmentées à partir de 1500 et 747 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 resp. IAM (base d'apprentissage et base de validation). Un second corpus, beaucoup plus important en taille, rassemble des textes collectés de la base Wikipédia française, resp. anglaise. À partir de ces deux corpus, nous avons généré trois configurations de vocabulaires et de modèles de langage.

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

4.4. Étape 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 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 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 $\hat{W}$ 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 précédemment, on arrive finalement à la formule ci-dessous qui régit l'étape de décodage. Dans cette formule, S représente la séquence d'observations extraites de l'image et $P(S|W)$ représente la vraisemblance que les caractéristiques 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.

$$\hat{W} = argmax_W P(S|W)P(W)^{\gamma} \beta^{taille(W)} \quad (2)$$

5. Évaluation

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. Pour les tests sur la base IAM, nous avons utilisé 1033 lignes de texte extraites de 336 images qui composent la base d'évaluation de la base IAM. L'optimisation des paramètres de décodage est réalisée sur la base de validation IAM.

Nous avons défini une première configuration de nos systèmes en combinant les ressources de la base RIMES, respectivement IAM, 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 tailles de lexiques, en ajoutant au lexique de la base RIMES resp. IAM différents ensembles constitués des mots les plus fréquents de la base Wikipédia (10 K, 20 K et 40 K mots les plus fréquents). Pour chaque taille de lexique, un modèle de langage est entraîné spécifiquement sur les corpus Wikipédia et RIMES resp. IAM.

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 Wikipédia pour déterminer des lexiques et leurs modèles de langage associés. Nous avons retenu les mêmes lexiques Wikipédia que pour le mode à lexiques fermés (10 K, 20 K et 40 K mots).

La figure 1 donne les taux de couverture (proportion des mots présents) du lexique et de la base RIMES resp. IAM pour les différents lexiques de mots et de syllabes extraits de Wikipédia. On voit que la base RIMES resp. IAM présente un taux de mots hors lexique assez important lorsqu'on travaille avec des lexiques de petite taille provenant de wikipédia. La figure 2 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. Cette remarque est valide pour les deux langues.

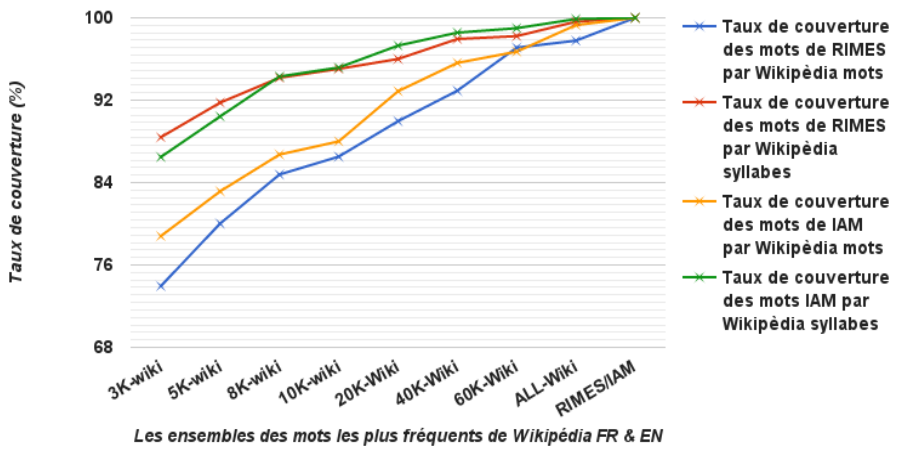

Figure 1. Taux de couverture de la base RIMES, respectivement IAM, par les lexiques de mots et de syllabes de Wikipédia français (FR) et anglais (EN)

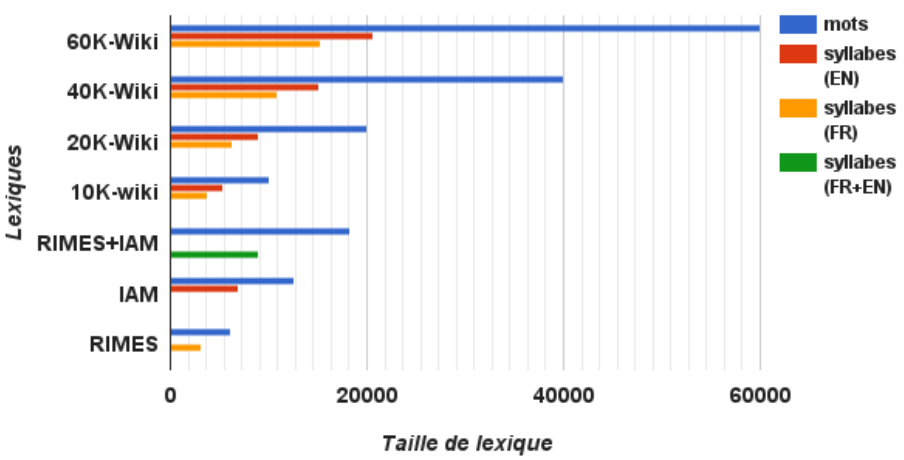

Figure 2. Taille des lexiques de syllabes dérivés des lexiques mots Wikipédia

Pour la méthode de syllabation, les valeurs raisonnables du seuil T peuvent être choisies entre [0.5, 1]. Entre ces deux valeurs extrêmes du seuil le choix d'une valeur optimale est crucial car lorsque T tend vers 1 l'algorithme tend à diviser systématiquement le mot requête en caractères, ce qui n'est pas le but recherché. Inversement, lorsque T se rapproche de 0.5, on a tendance à proposer

systématiquement une syllabation même si le mot inconnu possède une structure assez différente du mot le plus proche appartenant au lexique. Dans ce cas on a tendance à produire une syllabation erronée.

C'est le cas, par exemple, pour les vocabulaires qui ne font pas partie de la langue française parlée. Cette marge de syllabation par erreur doit être quantifiée par un expert linguiste. Cependant il nous faut optimiser la valeur de T pour trouver celle qui minimise à la fois le nombre des mots décomposés en caractères et le nombre de syllabations fausses. Une première possibilité consisterait à optimizer T par cros-validation. Une seconde approche consiste à optimiser T par rapport aux performances du système de reconnaissance. C'est cette seconde approche que nous avons retenue.

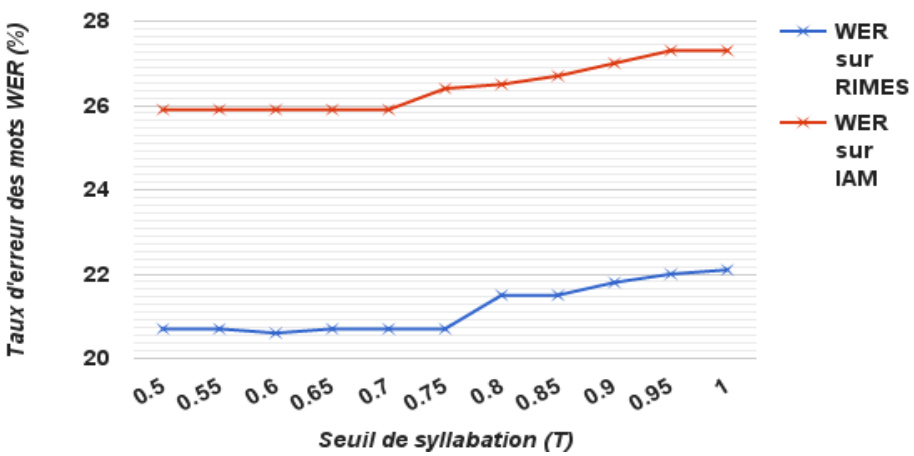

Figure 3. Taux d'erreur de mots (WER%) pour chaque valeur de T

La figure 3 présente l'évolution du taux d'erreur mots du système de reconnaissance sur les deux bases RIMES et IAM. On remarque un comportement assez similaire des deux systèmes avec un décrochage à partir d'une certaine valeur de T au-delà de laquelle le système de reconnaissance a un pouvoir de couverture du lexique amoindri et des performances qui se dégradent brutalement. Nous avons choisi $T = 0{,}6$ pour la base RIMES, et $T = 0{,}7$ pour la base IAM. Dans cette situation seulement 0,25 % des mots de la base RIMES ne sont pas syllabisés. Ce ne sont pas des mots de la langue française, comme par exemple la sequence *SXNHBOO*. On observe le même phénomène sur la base IAM, par exemple le mot "Lollobrigida" est décomposé en caractères. Pout $T = 0{,}7$ seulement 0,37 % des mots de la base IAM ne sont pas syllabisés, et décomposés en caractères par erreur.

Une analyse complémentaire de la méthode de syllabation est illustrée sur les figures 4 et 5, qui représentent les histogrammes du nombre de syllabes par mots pour les lexiques RIMES et IAM en fonction du seuil T. On peut remarquer que la majorité des mots se décompose en au plus 5 syllabes sur la base RIMES lorsque

$T = 0{,}6$. On remarque également qu'au-delà de 0,6 le nombre de mots anormalement décomposés en un nombre important de syllabes augmente, du fait que la méthode privilégie de plus en plus une décomposition des mots en caractères au-delà de cette valeur de seuil. On peut faire les mêmes observations sur la base IAM à partir de $T = 0{,}7$ (figure 5).

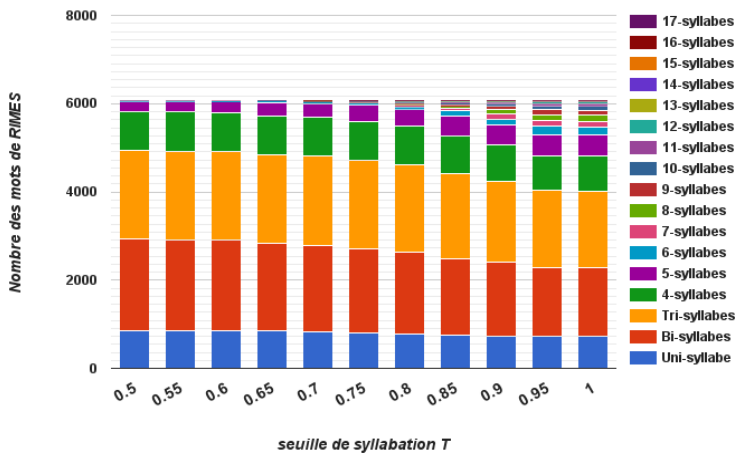

Figure 4. Histogrammes du nombre de syllabes par mots en fonction de T (base RIMES)

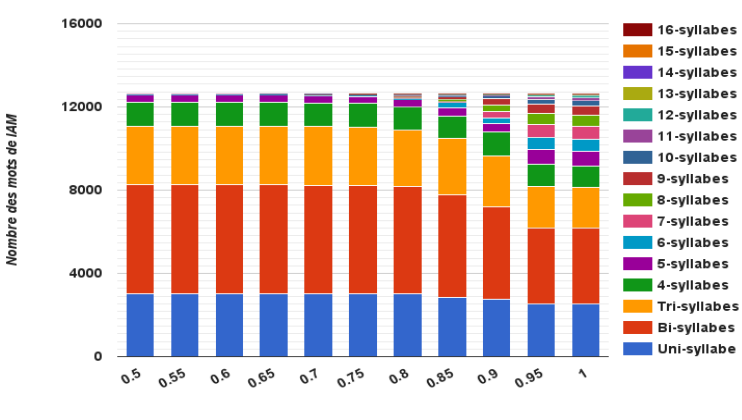

Figure 5. Histogrammes du nombre de syllabes par mots en fonction de T (base IAM)

Le taux d'erreur mot (*word error rate*, WER) est utilisé pour évaluer la performance de nos systèmes de reconnaissance. La figure 6 illustre le

comportement du système pour chaque modèle (caractères, syllabes et mots) et pour des configurations de lexique couvrant totalement, ou partiellement la base RIMES. Sur la figure on a représenté le WER pour des lexiques ayant un taux de couverture décroissant. À gauche on donne quatre résultats sur des lexiques incluant la base RIMES qui ont nécessairement un taux de couverture de 100 % des mots de la base de test, puis on donne quatre résultats pour des lexiques uniquement constitués à partir de wikipedia. Leur taille décroissante conduit à un taux de couverture décroissant (respectivement de 97,94 %, 96 %, 95,04 %, 94,18 % pour le modèle syllabique et de 92,92 %, 89,96 %, 86,52 %, 84,78 % pour le modèle mots).

On observe pour la configuration de lexique la plus spécialisée à la base RIMES (premier point à gauche), 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 en gardant un taux de couverture de 100 %. On observe même que les performances du modèle syllabique se dégradent moins lorsque la taille du lexique augmente, par rapport 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 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).

Sur la figure 6, les quatre tests réalisés en vocabulaire ouvert (lexiques Wikipédia seuls) montrent que le modèle syllabique obtient des performances égales ou supérieures au modèle mot. Les modèles sont voisins avec une préférence pour le modèle syllabique toutefois, lorsqu'on travaille avec un lexique de 40 K (dans ce cas les taux de couverture sont respectivement de 92,92 % pour le modèle mot, et 97,94 % pour le modèle syllabique). Dans les autres configurations, 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.

Globalement on observe le même comportement des trois modèles sur la base IAM (voir la figure 7). En lexique fermé de taille la plus faible (premier point à gauche sur la figure), le modèle syllabique obtient une performance de 25,9 % en WER très proche du modèle mot (24,5 % WER). Ce comportement se confirme pour des lexiques de taille supérieure. Les tests réalisés avec des vocabulaires ouverts, représentés par les trois points les plus à droite sur la figure 7 (avec des lexiques Wikipedia uniquement) montrent que le modèle syllabique atteint des performances égales ou supérieures au modèle mots. Les évolutions des performances pour les différentes configurations de lexiques sont identiques pour les deux langues.

On remarque la supériorité du modèle syllabique pour sa capacité à couvrir des mots hors lexique tout en gardant une complexité inférieure (nombre de syllabes inférieur au nombre de mots) et en atteignant des performances de reconnaissance supérieures au modèle mot.

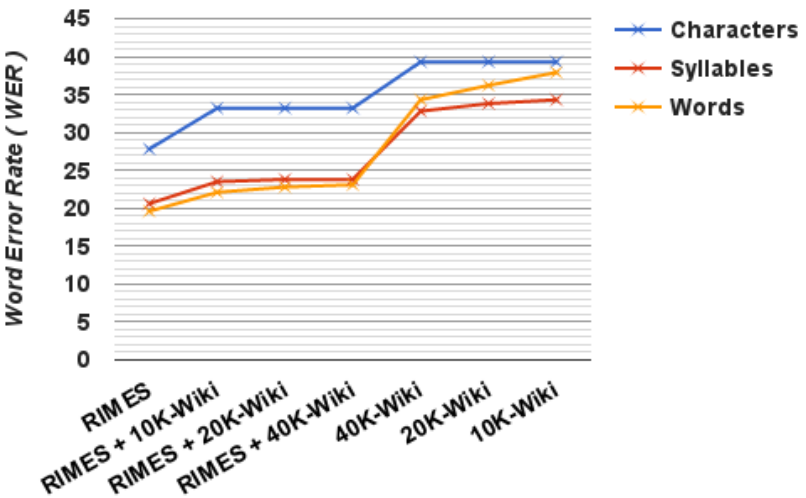

Figure 6. WER (%) des trois modèles, pour différentes tailles de lexiques de la base RIMES & Wikipèdia

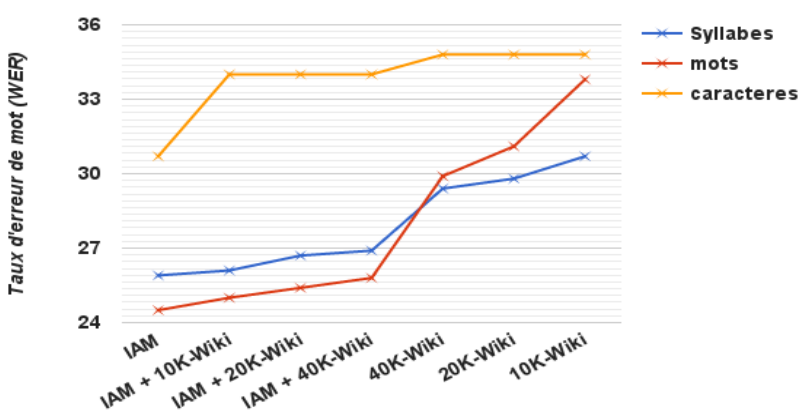

Figure 7. WER (%) des trois modèles, pour différentes tailles de lexiques de la base IAM & Wikipédia

6. Conclusion et perspectives

Dans cette étude nous avons proposé un modèle syllabique 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 de taille réduite. 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. Les expérimentations réalisées pour le français et pour l'anglais nous amènent aux mêmes conclusions, ce qui renforce l'intérêt et la généralité de l'approche. Les perspectives à court terme de ce travail seront d'examiner le comportement de cette approche syllabique couplée avec des modèles optiques plus performants tels que des réseaux de neurone récurrents qui sont actuellement les systèmes de l'état de l'art. Ces modèles optiques discriminant sont en effet capables de modéliser le contexte sur des fenêtres de temporelles de plusieurs caractères, ce que les modèles HMM ne peuvent pas faire.

Pour générer le modèle syllabique, nous nous sommes appuyé sur le dictionnaire *lexique3* et sur le dictionnaire *English Language Hyphenation Dictionary*, qui proposent tout deux une modélisation syllabique orthographique du français resp. de l'anglais. 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 Ryst et M. Christophe Coupeur qui nous ont offert leurs conseils en tant que spécialistes linguistes.

Bibliographie

AbdulKader A. (2008). A two-tier arabic offline handwriting recognition based on conditional joining rules. In *Arabic and Chinese Handwriting Recognition*(pp. 70-81). Springer Berlin Heidelberg. , pp. 70-81.

Ait-Mohand K., Paquet T., & Ragot N. (2014). Combining structure and parameter adaptation of HMMs for printed text recognition. *IEEE transactions on pattern analysis and machine intelligence*, 36(9), 1716-1732.

Angoujard J. P. (1997). *Théorie de la syllabe : rythme et qualité*. CNRS.

Bartlett S., Kondrak G., & Cherry C. (2009, May). On the syllabification of phonemes. *Proceedings of Human Language Technologies, The 2009 Annual Conference of the*

North American Chapter of the Association for Computational Linguistics, pp. 308-316. Association for Computational Linguistics.

BenZeghiba M. F., Louradour J., & Kermorvant C. (2015, August). Hybrid word/Part-of-Arabic-Word Language Models for arabic text document recognition. *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*, pp. 671-675. IEEE.

Brakensiek A., Rottland J., & Rigoll G. (2002). Handwritten address recognition with open vocabulary using character n-grams. *Frontiers in Handwriting Recognition, 2002. Proceedings. Eighth International Workshop*, pp. 357-362. IEEE.

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

Cirera N., Fornés A., & Lladós J. (2015, August). Hidden Markov model topology optimization for handwriting recognition. *Document Analysis and Recognition (ICDAR), 2015 13th International Conference*, pp. 626-630. IEEE.

Creutz M., Hirsimäki T., Kurimo M., Puurula A., Pylkkönen J., Siivola V., & Stolcke A. (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.

Doetsch P., & Ney H. (2013, August). Improvements in rwth''s system for off-line handwriting recognition. *2013 12th International Conference on Document Analysis and Recognition*, pp. 935-939. IEEE.

Feild J. L., Learned-Miller E. G., & Smith D. A. (2013, August). Using a probabilistic syllable model to improve scene text recognition. *2013 12th International Conference on Document Analysis and Recognition*, pp. 897-901. IEEE.

Flipo D., Gaulle B., et Vancauwenberghe K. (1994). Motifs français de césure typographique. *Cahiers gutenberg n°18*.

Gorski N., Anisimov V., Augustin E., Baret O., Price D., & Simon J. C. (1999, September). A2ia check reader: A family of bank check recognition systems. In *Document Analysis and Recognition, 1999. ICDAR''99. Proceedings of the Fifth International Conference,* pp. 523-526. IEEE.

Grosicki E., & El-Abed H. (2011, September). Icdar 2011-french handwriting recognition competition. *2011 International Conference on Document Analysis and Recognition*, pp. 1459-1463. IEEE.

Hamdani M., Mousa A. E. D., & Ney H. (2013, August). Open vocabulary Arabic handwriting recognition using morphological decomposition. *2013 12th International Conference on Document Analysis and Recognition*, pp. 280-284. IEEE.

Hearing (2000). *for Language Speech, T.I.,. Grady Ward''s Moby.* Marti U.V., Bunke H., 2002. http: //icon.shef.ac.uk/Moby/. [Online; accessed 21-Dec-2015].

Hindson M. (2016). http://hindson.com.au/info/free/free-english-language-hyphenation-dictionary/

Hsu B. J. P. (2009). *Language modeling for limited-data domains*, Doctoral dissertation, Massachusetts Institute of Technology.

Koerich A. L., Sabourin R., & Suen C. Y. (2003). Large vocabulary off-line handwriting recognition: A survey. *Pattern Analysis & Applications*, 6(2), 97-121.

Kozielski M., Matysiak M., Doetsch P., Schlöter R., & Ney H. (2014, September). Open-lexicon Language Modeling Combining Word and Character Levels. *Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference*, pp. 343-348. IEEE.

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

New B., Pallier C., Brysbaert M., & Ferrand L. (2004). Lexique 2: A new French lexical database. *Behavior Research Methods, Instruments, & Computers*, 36(3), 516-524.

Pantke W., Märgner V., Fecker D., Fingscheidt T., Asi A., Biller O., & Yehia M. (2013, January). HADARA–A software system for semi-automatic processing of historical handwritten Arabic documents. *Archiving Conference Conference* (vol. 2013, n° 1, pp. 161-166). Society for Imaging Science and Technology.

Plötz T., & Fink G. A. (2009). Markov models for offline handwriting recognition: a survey. *International Journal on Document Analysis and Recognition (IJDAR)*, 12(4), 269-298.

Prasad R., Saleem S., Kamali M., Meermeier R., & Natarajan P. (2008, December). Improvements in hidden Markov model based Arabic OCR. *Pattern Recognition, ICPR 2008, 19th International Conference*, pp. 1-4. IEEE.

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

Roekhaut S. B. S., & Beaufort R. (2012). *Syllabation graphémique automatique à l'aide d'un dictionnaire phonétique aligné*.

Rosenfeld R. (1995). *Optimizing lexical and ngram coverage via judicious use of linguistic data*. Computer Science Department.

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

Swaileh W., Mohand K. A., & Paquet T. (2015, August). Multi-script iterative steerable directional filtering for handwritten text line extraction. *Document Analysis and Recognition (ICDAR), 2015 13th International Conference*, pp. 1241-1245. IEEE.

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

Zimmermann M., & Bunke H. (2002). Hidden Markov model length optimization for handwriting recognition systems. *Frontiers in Handwriting Recognition, 2002.*