Décodeur neuronal pour la transcription de documents manuscrits anciens

Adeline GRANET[1], Emmanuel MORIN[1], Harold MOUCHÈRE[1],
Solen QUINIOU[1], Christian VIARD-GAUDIN[1]
(1) LS2N UMR 6004, 2 rue de la Houssinière, 44322 Nantes, France
prénom.nom@ls2n.fr

RÉSUMÉ

L'absence de données annotées peut être une difficulté majeure lorsque l'on s'intéresse à l'analyse de documents manuscrits anciens. Pour contourner cette difficulté, nous proposons de diviser le problème en deux, afin de pouvoir s'appuyer sur des données plus facilement accessibles. Dans cet article nous présentons la partie décodeur d'un encodeur-décodeur multimodal utilisant l'apprentissage par transfert de connaissances pour la transcription des titres de pièces de la Comédie Italienne. Le décodeur transforme un vecteur de n-grammes au niveau caractères en une séquence de caractères correspondant à un mot. L'apprentissage par transfert de connaissances est réalisé principalement à partir d'une nouvelle ressource inexploitée contemporaine à la Comédie-Italienne et thématiquement proche ; ainsi que d'autres ressources couvrant d'autres domaines, des langages différents et même des périodes différentes. Nous obtenons 97,27% de caractères bien reconnus sur les données de la Comédie-Italienne, ainsi que 86,57% de mots correctement générés malgré une couverture de 67,58% uniquement entre la Comédie-Italienne et l'ensemble d'apprentissage. Les expériences montrent qu'un tel système peut être une approche efficace dans le cadre d'apprentissage par transfert.

ABSTRACT

Neural decoder for the transcription of historical handwritten documents.

The lack of data can be an issue at the beginning of a study on new historical handwritten documents. To solve this issue, we present the decoder part of a multimodal approach based on transductive transfer learning for transcripting play titles of the Italian Comedy.

MOTS-CLÉS : modèle neuronal, apprentissage par transfert, transcription, Comédie Italienne.

KEYWORDS: neural model, transfer learning, transcription, Italian Comedy.

1 Introduction

La préservation de notre héritage culturel passe par la numérisation de documents historiques. La consultation de ces documents nécessite leur indexation afin de pouvoir y accéder efficacement. Pour cela, de nombreuses études interdisciplinaires faisant intervenir conjointement les sciences du traitement automatique des langues, de la reconnaissance des formes dans les documents et de la recherche d'information se sont développées.

Avec les documents anciens, l'un des enjeux importants concerne les évolutions orthographiques au fil du temps. Le problème de la normalisation de ces fluctuations est toujours délicat (Garrette

& Alpert-Abrams, 2016; Bollmann *et al.*, 2017). En reconnaissance d'objets, les enjeux sont plus nombreux et divers : détection et segmentation des lignes, détection automatique de mots-clés ou encore reconnaissance d'écriture. Ces dernières années, les compétitions tournant autour des documents historiques se multiplient (Cloppet *et al.*, 2016; Pratikakis *et al.*, 2016; Sanchez *et al.*, 2017). Les systèmes doivent s'adapter au support des documents, au niveau de détérioration ou encore au style de l'écriture. Tous ces facteurs ont un fort impact sur les systèmes.

En reconnaissance d'écriture, les réseaux les plus performants sont construits à partir de réseaux de neurones profonds, et plus récemment, ils intègrent des modèles d'attention comme (Bluche *et al.*, 2017). Les systèmes de reconnaissances sont construits avec des réseaux multi-dimensionnels utilisant des cellules de type *Long Short Term Memory* (MDLSTM), ou encore des réseaux récurrents à convolution (CRNN) associés à des réseaux bidirectionnels (BLSTM) (Granell *et al.*, 2018). Ces approches permettent d'utiliser tout le contexte disponible sur les images. Cependant, la phase de décodage de ces séquences dépend directement de la taille du vocabulaire utilisé pour construire un modèle de langage ou un dictionnaire. Lorsqu'une grande quantité de mots est hors-vocabulaire, les résultats se dégradent. Le problème majeur de ces réseaux est la quantité de ressources nécessaire à leur apprentissage.

L'étude envisagée concerne la Comédie Italienne pour laquelle aucune ressource annotée n'est disponible. Il n'est donc pas possible de mettre en œuvre directement un tel type de réseau.

L'apprentissage transductif par transfert de connaissance est une approche intéressante dans le cas où il y a un manque, voir une absence de données pour réaliser l'apprentissage d'un système. En effet, cette méthode consiste à utiliser différentes sources de données pour l'apprentissage d'un système dédié à une tâche, et appliquée sur des données différentes (Pan & Yang, 2010). Il est donc possible pour nous à partir de différentes données connues d'annoter des données inconnues. Ce procédé est utilisé pour alimenter les systèmes d'apprentissage gourmands dans différents domaines tels que la détection de mot automatique dans les documents historiques (Lladós *et al.*, 2012) ou pour les modèles multimodaux de traduction (Nakayama & Nishida, 2017). Notre solution est d'utiliser l'apprentissage par transfert de connaissances pour faire de la reconnaissance d'écriture sur ces nouveaux documents.

Les méthodes standards en traduction automatique utilisent des systèmes de type encodeur-décodeur à partir de réseaux neuronaux récurrents (Cho *et al.*, 2014a). Le premier élément encode une donnée source d'un langage en un vecteur, et le second élément décode la séquence dans une langue cible. Vinyals *et al.* (2015) a proposé un générateur de légendes pour les images constitué de deux réseaux : un réseau à convolution (CNN) pré-entraîné encodant une image dans un vecteur de taille fixe, et un réseau LSTM générant la description de l'image. Nous avons extrapolé cette approche pour l'appliquer à un système de reconnaissance d'écriture. La représentation intermédiaire retenue en sortie de l'encodeur utilise un espace explicite fondé sur les n-grammes.

Nous souhaitons en particulier mettre en place un système de type encodeur-décodeur afin de réaliser un apprentissage par transfert à deux niveaux : l'un pour encoder les images de mots et l'autre pour décoder vers le texte. Cette étude préliminaire vise à rendre compte de l'efficacité du décodeur à générer des mots issus du vocabulaire de la Comédie-Italienne de la représentation intermédiaire résultant de l'encodage de l'image.

2 Modèles pour l'apprentissage par transfert de connaissances

Nous souhaitons créer un système de reconnaissance d'écriture manuscrite pour des documents multilingues anciens à partir de ressources différentes en termes de langue et d'époque. En se basant sur les travaux réalisés pour la génération de description d'images, le modèle que nous proposons se décompose en deux parties complémentaires (voir Figure 1) comme (Vinyals *et al.*, 2015). La première partie a pour but d'encoder l'image d'un mot et de la convertir en un vecteur. La seconde partie, quant à elle, doit décoder ce vecteur pour en générer une séquence de caractères. L'originalité de notre approche réside dans le fait d'utiliser un vecteur de n-grammes comme pivot du système et de supprimer la notion de temporalité entre les caractères d'un mot. Ce vecteur permet d'encoder les informations dans un espace non-latent qui est transférable tant que les données d'apprentissage et les données de cible de transfert partagent le même alphabet. Une telle approche favorise un apprentissage indépendant des deux parties du système : le modèle optique et le modèle de langage.

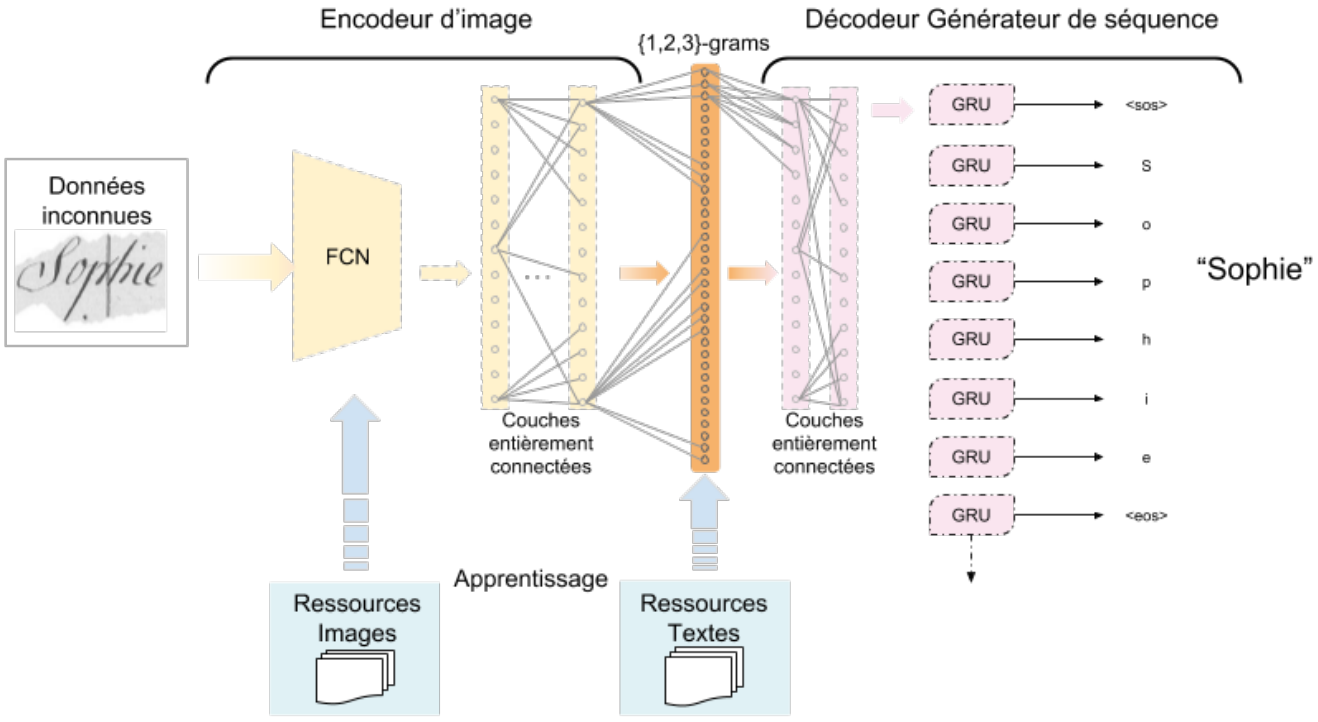

FIGURE 1 – Vue d'ensemble de l'architecture de l'encodeur-décodeur

2.1 Encodeur d'images

La tendance en apprentissage par transfert est à l'utilisation de réseaux pré-entraînés sur des images naturelles pour en extraire les caractéristiques. Ces images sont rarement en niveaux de gris contrairement aux images que nous exploitons. Nous avons donc choisi de définir et d'entraîner notre propre encodeur selon l'architecture suivante :

- un réseau entièrement à convolution (FCN) pour l'extraction des caractéristiques ;
- deux couches entières connectées avec une fonction d'activation de type ReLU (Nair & Hinton, 2010) et 1 024 neurones par couche ;
- une dernière couche entièrement connectée avec une fonction d'activation de type *Sigmoid* et $L + 1$ neurones où L correspond aux nombres de n-grammes estimés, et un neurone

supplémentaire comme joker si le n-gramme souhaité est absent de la liste.
Le FCN extrait et résume les caractéristiques des images prises en entrée. La dernière couche utilisant une *Sigmoid* permet d'obtenir une probabilité pour chaque n-gramme disponible et indépendamment des autres là où un *Softmax* normaliserait l'ensemble des sorties pour obtenir une somme égale à 1.

2.2 Décodeur de n-grammes

Dans le domaine du traitement du langage, il est courant d'utiliser des structures de type encodeur-décodeur prenant en entrée une séquence à partir de laquelle une autre séquence est générée, sans pourtant avoir la même longueur ou les mots dans le même ordre comme c'est le cas en traduction (Cho *et al.*, 2014b). Cette solution semble correspondre aux conditions imposées par l'apprentissage par transfert de connaissances et les données utilisées. Nous choisissons l'implémentation suivante :

— une première couche entière connectée avec une activation de type ReLU ;
— une couche récurrente de type GRU ;
— une dernière couche entièrement connectée avec une fonction d'activation de type *Softmax*.

Cette architecture volontairement simple pourrait l'être encore plus si nous utilisions une couche de plongement lexical pré-entraîné. Mais à notre connaissance, il n'existe pas de plongements lexicaux multilingues pour les n-grammes disponibles. Pour obtenir une séquence, nous utilisons une couche récurrente qui va générer de la temporalité. La dernière couche doit fournir un caractère jusqu'à ce que le symbole de fin de mot soit émis. Dans les modèles encodeur-décodeur notamment utilisés en traduction, la partie décodeur utilise un réseau bi-directionnel pour prendre en compte toutes les informations contenues dans une phrase à travers une matrice. Or, nous utilisons un vecteur comme entréece qui signifie que l'information temporelle spécifiant la position d'un n-gramme par rapport à un autre disparu. Un réseau bi-directionnel ne sera donc pas utile dans notre cas.

3 Ressources utilisées

Nous cherchons à réaliser une solution de reconnaissance d'écriture manuscrite sur les registres de la Comédie Italienne (RCI), pour lesquels nous avons peu de données annotées. Nous présentons les ressources utilisées pour nos expérimentations.

3.1 Données cibles : les registres de la Comédie Italienne

Ces documents sont des registres financiers de la Comédie-Italienne datant du XVIII^{e} siècle avec environ 28 000 pages. Voici quelques observations que nous avons pu effectuées :

— la langage évolue de l'Italien au Français, au début du siècle les acteurs eux-même rédigeaient les documents, avant d'engager un caissier ;
— la présentation des comptes journaliers change également au cours du siècle mais tout en préservant la présence des informations ;
— le style de l'écriture est très variable entre le début du siècle et 1730, avant de se stabiliser grâce au caissier.

La figure 2 présente un format de page d'un compte journalier où l'on distingue les informations suivantes : date du jour, titres des pièces qui ont été jouées ce soir-là, recettes et des notes (dans la colonne de gauche) et dépenses et liste des acteurs et actrices (dans la colonne de droite).

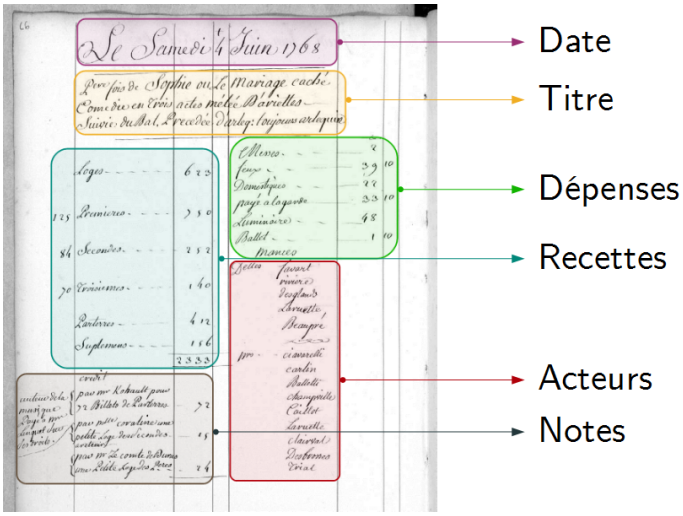

FIGURE 2 – Exemple d'une page journalière des registres financiers de la Comédie-Italienne avec l'identification des zones d'informations.

Dans la suite de nos expérimentations, nous nous concentrons sur la zone de titre. Cette zone contient une liste succincte des pièces jouées, qui peut être complétée par des informations indiquant si c'était une première de la pièce ou d'un acteur... La zone de titre de la figure 2 explique que la pièce « Sophie ou le mariage caché » a été jouée et que c'était une comédie en trois actes précédée des pièces « Arlequin toujours arlequin » et « mêlée d'ariettes ». Il faut également noter qu'un titre peut être écrit de plusieurs manières et reste principalement constitué d'entités nommées comme « Raton et Rosette » ou « Zémire et Azor ».

Pour le décodeur que nous réalisons, nous devons faire attention au langage et au style de l'écriture. En comparant le style contemporain de l'écriture avec l'historique, nous avons noté quelques différences dont la présence de caractères spéciaux comme la forme longue du 's' ; une évolution de la langue qui a converti les 'y' de fin de mot en 'i' ; ou encore le 'i' et 'j' qui ne se différenciaient pas.

Grâce à un site d'annotation participatif dédié à ces données de la Comédie Italienne, nous avons collecté 971 lignes de titres ce qui correspond à 4 938 mots au total. Nous utilisons l'ensemble des mots contenu dans ces lignes pour constituer la base de test pour le décodeur générateur de séquence, soit 1 431 mots différents. Cela reste un sous-ensemble du vocabulaire contenu dans l'ensemble des 28 000 pages de la Comédie-Italienne. Il est important pour l'apprentissage que nous sélectionnions des ressources avec des caractéristiques similaires à celles de la Comédie Italienne pour avoir un système efficace.

3.2 Les ressources pour l'apprentissage

Une nouvelle ressource. Pour alimenter le modèle de langage, nous pallions le manque de données de la ressource RCI en intégrant des données textuelles additionnelles provenant de 23 œuvres traitant de la Comédie Italienne, publiées au XVIII^{e} siècle et disponibles sur *Google Livre*. Parmi ces œuvres, nous trouvons des scripts bilingues (en Italien et en Français), des répertoires d'œuvres, des livres d'anecdotes sur le théâtre italien... Les œuvres obtenues ont été nettoyées pour supprimer le bruit induit par la numérisation, comme la structure du texte et les caractères spéciaux. Cette nouvelle ressource, appelé GCI, a l'avantage de posséder un vocabulaire proche des données manuscrites que nous étudions.

Les ressources existantes. Pour l'apprentissage par transfert de connaissances de l'encodeur d'images, nous avons sélectionné un ensemble de ressources manuscrites ayant au moins un point commun avec les données de la Comédie Italienne :

— RIMES (RM) (Reconnaissance et Indexation de données Manuscrites et de facsimilés) est une base française de demandes administratives (Grosicki & El-Abed, 2011) ;
— Los Esposalles (ESP) est construit sur des registres de mariages espagnols du XV^{e} au XVII^{e} siècle (Romero *et al.*, 2013) ;
— Georges Washington (GW) est une base en anglais créée à partir de 20 lettres de correspondance (Fischer *et al.*, 2012) ;
— Wikipedia français (Wiki), utilisée et distribuée par (Bojanowski *et al.*, 2017), est une ressource contenant tous les mots qui ont une fréquence supérieure à 5 dans Wikipedia. Nous en sélectionnons aléatoirement 30 000 mots.

	GCI	RM	ESP	GW	Wiki	RCI
Apprentissage	26 573	4 477	2 565	660	24 456	0
Validation	2 953	1,578	629	521	3 843	0
Test	0	1 627	629	431	1 928	1 431

TABLE 1 – Nombre de mots uniques des ressources exploitées.

La table 1 donne la taille du vocabulaire de chaque ressource. L'objectif est de transcrire les registres de la Comédie Italienne contenant 1 431 mots différents. Le taux de mots hors-vocabulaire par rapport aux autres ressources varie de 34 à 99 %. GCI obtient le taux le plus faible, ce qui confirme son choix, alors que le taux de mots hors-vocabulaire, par rapport à la ressource RM, est de 87,47 %.

Pour l'ensemble des ressources, nous avons remplacé les caractères accentués par leur forme simple, comme par exemple [é,è, ê,ë] par le caractère "e", ainsi que les formes spéciales comme la forme longue du "s", typique du XVIII^{e}, siècle en sa forme courte. Nous conservons également la casse. Pour les livres GCI, les lignes de texte ont été coupées sur les espaces et les signes de ponctuation. Pour supprimer les séquences de caractères qui ne sont pas des mots, nous avons uniquement conservé les séquences ayant une fréquence supérieure à 2 dans l'ensemble de la ressource.

Comme (Bengio & Heigold, 2014), nous utilisons les n-grammes de caractères pour représenter les mots. L'élément pivot de notre encodeur-décodeur est un vecteur de n-grammes de caractères. Initialement, les auteurs ont sélectionné les 50 k n-grammes les plus fréquents. Nous calculons tous les n-grammes possibles avec une longueur maximum de 3 sur l'ensemble des ressources d'apprentissage et en ajoutant les symboles de début *[* et fin de mot *]*. Par exemple, la décomposition du mot *Sophie* de la figure 1 est {[S,o,p,[So,op,Sop,...,ie],e]} soit un total de 19 n-grammes et d'une manière générale $3n + 1$ n-grammes de longueur maximale 3 pour un mot de n caractères. Pour GCI, nous filtrons les n-grammes ayant une occurrence strictement supérieure à 5 et ceux présents dans au moins deux ressources différentes, et un joker est ajouté pour remplacer les n-grammes non-sélectionnés. Il en résulte un nombre total de 12 500 n-grammes.

Pour l'entrée du décodeur, le vecteur de n-grammes est construit par normalisation de la fréquence de chaque n-gramme présent dans le mot. Cela permet de conserver une information sur la taille du mot et de compenser la séquentialité supprimée. Pour générer les séquences de sorties, nous avons 79 neurones représentant toutes les lettres minuscules, majuscules, les chiffres, les symboles de ponctuations dont l'espace et les symboles de début et fin de mot. Un dernier neurone est ajouté pour permettre au réseau de ne plus répondre de caractères après la fin du mot.

4 Paramétrage

Comme la taille de nos ressources varie de 660 à environ 25 000 mots, nous avons entraîné le décodeur avec une ou plusieurs ressources hormis pour GW (comme la taille est trop petite nous combinons systématiquement cette ressource à d'autres). Pour éviter le sur-apprentissage dans notre réseau, nous utilisons la méthode d'arrêt prématuré qui consiste à stopper le réseau quand au bout de cinq itérations la fonction coût sur la base de validation ne décroit plus. La structure du décodeur comprend 1 024 neurones dans la première couche entière connectée pour extraire les caractéristiques, suivi de 500 neurones cachés dans la couche GRU et enfin 79 neurones avec *Softmax* en fonction d'activation. Le taux d'apprentissage fixé à 0,0001 est géré automatiquement grâce à la fonction Adam. Pour pouvoir générer une séquence, nous fixons la longueur maximum à 50 caractères. Le réseau crée ainsi des séquences de longueurs différentes sans contrainte.

Nous évaluons notre système grâce à un taux de reconnaissance sur les caractères (TRC) et sur les mots (TRM). Le TRC est défini par $(N - (Ins + Subs + Dels))/N$ où N représente le nombre de caractères dans le mot de référence, $Subs$ le nombre de caractères substitués, $Dels$ le nombre de caractères supprimés et Ins le nombre de caractères insérés. Le TRM correspond au rappel, c'est-à-dire, le nombre de mots correctement reconnus par rapport au nombre de mots dans l'ensemble de référence. Nous calculons ces taux selon quatre options : i) avec et sans dictionnaire pour aider au décodage de la séquence et ii) avec et sans majuscule pendant le décodage. Le dictionnaire est construit sur le vocabulaire des ensembles d'apprentissage et de validation de l'ensemble des ressources. Cela nous donne un dictionnaire avec 39 051 entrées. Nous calculons également la couverture lexicale fournie par chaque ensemble d'apprentissage par rapport au test. Cela nous donne une borne haute pour le taux de reconnaissance de mots à atteindre avec le dictionnaire. Cette couverture lexicale correspond au nombre de mots communs à l'ensemble d'apprentissage et à celui de test, divisé par la taille du vocabulaire de l'ensemble de test.

5 Expériences

La table 2 montre les résultats obtenus pour la génération de séquences. Nous avons réalisé trois types d'expériences :

1. avec la même ressource pour l'apprentissage et le test ;
2. en ajoutant d'autres ressources pour l'apprentissage ;
3. en utilisant uniquement des ressources différentes de l'ensemble de test, pour l'apprentissage, ce qui correspond à l'apprentissage par transfert.

Même si le but de notre approche est bien de pouvoir décoder les mots de la Comédie-Italienne, nous présentons aussi les résultats obtenus sur RM et ESP. Cela nous semble intéressant de pouvoir observer si la méthode est applicable pour différents types de ressources.

Pour commencer, nous utilisons uniquement les unigrammes pour représenter un mot et la même ressource pour l'apprentissage et le test. Nous constatons avec RM, comme avec GCI, que les résultats sont meilleurs lorsque l'on considère les n-grammes de caractères. Sur GCI, le TRM augmente de 70 % sans l'utilisation du dictionnaire, tout en dépassant la couverture lexicale. Notons également que seul 3 % des caractères sont mal reconnus. Ces résultats corroborent les études utilisant les trigrammes comme (Vania & Lopez, 2017). Pour la suite des expériences, nous utilisons uniquement les vecteurs

de n-grammes.

Sur l'ensemble des expériences, nous obtenons globalement les mêmes résultats avec ou sans majuscules : $\pm$ 0.07 % pour le TRC, $\pm$ 1 % pour le TRM et $\pm$ 1.05 % avec le dictionnaire. Les erreurs de caractères du système ne sont donc pas uniquement des minuscules prédites en majuscules. L'utilisation du dictionnaire fait chuter les performances du décodeur. Nous remarquons que lorsque la couverture lexicale est supérieure à 20 %, le TRM reste inférieur à celle-ci. Le dictionnaire induit en erreur le décodeur quand il génère une séquence correcte et que le mot correspondant n'appartient pas au dictionnaire. Cependant, il est également capable d'aider le décodeur à se rapprocher du mot de référence même si la forme exacte n'est pas contenue dans le dictionnaire, et qu'en plus, les données d'apprentissage et de test utilisées sont d'époques et de langues différentes. Seulement, ces cas sont trop rares pour améliorer le TRC.

Test sur RCI Les résultats obtenus pour les TRC et TRM, lorsque GCI est combiné avec d'autres ressources, sont très similaires aux résultats obtenus sur GCI seul (Expérience 2 *vs.* Expérience 3, 4, et 5). Dans ce cas précis, l'augmentation de la quantité de données n'a pas un impact flagrant. Notons quand même que les meilleurs résultats sont obtenus en utilisant toutes les ressources en apprentissage. Dans le cas de l'apprentissage par transfert par rapport au domaine, la couverture lexicale est très basse, puisqu'elle est autour de 15 %. Cependant, le décodeur est capable d'atteindre 30,42 % de TRM en utilisant uniquement RM qui est en français contemporain, et 41,40 % en utilisant Wikipedia. Les résultats avec Wikipedia sont similaires à ceux obtenus en utilisant toutes les ressources avec une quantité inférieure de données. Finalement, chercher de nouvelles ressources encore inexploitées sur la Comédie Italienne est une approche intéressante : cela nous permet d'avoir un TRM supérieur de 20 % à la couverture lexicale. Parmi ces mots inconnus mais bien reconnus, nous retrouvons des abréviations telles que « arleq. » au lieu d'« Arlequin ».

Test sur RIMES Les résultats avec RM sont intéressants car c'est la seule ressource que nous utilisons qui est en français contemporain. Dans le cadre de l'apprentissage par transfert, sans RM dans les ressources d'apprentissage, les TRC et TRM atteignent les résultats obtenus sur RM seul. De plus, nous constatons qu'ils dépassent largement la couverture lexicale de 20 %. Lorsque nous travaillons avec des données historiques appliquées sur des données modernes mais partageant la même langue, la couverture lexicale est plus élevée que lorsque l'on utilise RM sur GCI. Ainsi, l'orthographe historique est plus facilement applicable sur du moderne pour le décodeur.

Test sur Los Esposalles Le vocabulaire de Los Esposalles est principalement construit à partir d'entités nommées. Ceci explique la couverture lexicale nulle dans le cadre du apprentissage par transfert. Cependant, le TRC est supérieur à 90 %, et le TRM dépasse la couverture lexicale de 52,91 %. Nous n'avons pas expérimenté l'utilisation seule de la base d'apprentissage de référence de cette ressource car elle est trop petite.

Analyse des erreurs La table 3 montre quelques erreurs récurrentes que nous avons pu constater. Parmi les erreurs observées, nous constatons que les mots ayant des caractères répétés, comme « cavalcade » et « clemence », posent plus de difficulté au système pour générer les caractères qui s'intercalent avec le caractère répété : il propose « cacaadade » et « ccceeene », respectivement. Cela représente environ 5% des erreurs constatées dans les expériences sur RCI. Une autre erreur commune

Test	Apprentissage	Expe. Id	N-grams	% Couv. lexicale	Sensible à la casse			Insensible à la casse		
					TRC	TRM	TRM dict.	TRC	TRM	TRM dict.
RCI	GCI	1	1	65,57	69,27	14,54	10,83	69,28	14,54	11,33
	GCI	2	1,2,3		97,10	86,22	39,30	97,17	86,26	40,14
	GCI+RM	3	1,2,3	67,58	**97,27**	**86,57**	**39,23**	97,27	86,57	40,07
	GCI+ESP	4	1,2,3	65,83	96,96	85,87	39,09	96,96	85,87	40,07
	GCI+ESP+GW+RM	5	1,2,3	67,65	95,85	79,65	38,25	97,42	87,13	39,16
	RM	6	1,2,3	14,52	79,70	30,42	17,27	79,75	30,49	17,76
	RM+ESP+GW	7	1,2,3	23,39	83,68	40,21	23,99	83,74	40,42	24,41
	Wiki	8.1	1,2,3	0.0	**87,32**	**41,40**	**25,24**	87,44	42,22	27,76
	Wiki 300k	8.2	1,2,3	0.0	92.80	55.94	29.37	93.00	57.27	31.61
RM	RM	9	1	75,09	83,97	43,07	28,49	83,98	43,07	28,98
	RM	10	1,2,3		94,72	79,50	37,78	94,74	79,63	37,84
	GCI+RM	11	1,2,3	83,83	98,25	92,0	40,49	98,25	92,0	40,55
	GCI+ESP+GW+RM	12	1,2,3	83,95	96,22	80,73	39,14	96,22	80,74	39,45
	GCI	13	1,2,3	58,55	**95,51**	**81,53**	**38,58**	95,51	81,53	38,58
	GCI+ESP	14	1,2,3	59,04	95,46	80,61	38,15	95,46	80,61	38,15
	Wiki	15	1,2,3	0.0	90,36	67,57	35,20	90,43	67,69	36,12
ESP	GCI+ESP	16	1,2,3	85,94	98,57	91,11	56,51	98,57	91,11	57,14
	GCI+ESP+GW+RM	17	1,2,3	86,10	98,40	90,79	57,62	98,40	90,79	57,78
	GCI	18	1,2,3	15,96	91,68	**65,87**	44,76	91,69	65,87	44,76
	RM	19	1,2,3	7,27	72,83	18,25	12,70	72,86	18,25	12,86
	GCI+RM	20	1,2,3	17,37	**92,05**	64,13	**46,51**	92,06	64,13	47,14
	GCI+RM+GW	21	1,2,3	17,69	91,68	64,60	44,60	91,70	64,76	45,07
	Wiki	22	1,2,3	0.0	84,52	34,28	32,38	84,71	35,08	34,12

TABLE 2 – Résultats pour la génération de séquence expérimentant l'apprentissage par transfert de connaissances : les TRC et TRM sont calculés avec ou sans l'utilisation du dictionnaire sur les différentes ressources.

est la permutation entre deux caractères comme avec « suitte » qui double un caractère à la place d'un autre. Cette erreur est la plus commune, elle couvre 79% des erreurs observées. Un dernier exemple avec « [ollat » pour « Soldat » qui apparaît lorsque le décodeur prédit deux symboles début de mot consécutifs, le second remplaçant le premier caractère du mot. Suivant la taille de la ressource utilisée pour l'apprentissage, ce type d'erreur représente entre 7% et 25% des erreurs, respectivement avec GCI, et RM seul.

Type d'Erreur	Expe. Id	Mot d'origine	Mot reconstitué
Caract. multiplié	4 3	cavalcade clemence	cacaadade ccceeene
Caract. interverti	6	suitte	usitte
Caract. de début	5 6	[diverstissemens] Soldat	ddevvestissemens] [ollat

TABLE 3 – Examples d'erreurs réalisées par le décodeur.

6 Conclusion

Dans cet article, nous nous sommes intéressés à la mise en place d'un apprentissage par transfert pour palier un manque de vérité terrain pour un système de reconnaissance d'écriture manuscrite. Contrairement aux travaux état-de-l'art, nous commençons par déconstruire la séquence initiale pour passer par une représentation intermédiaire robuste pour absorber les mots hors-vocabulaires (encodage), puis une séquence plausible est générée (décodage). Nos résultats montrent que l'approche est opérationnelle au niveau mot. Nous obtenons ainsi des TRC supérieurs à 90 % et des TRM dépassant la couverture lexicale estimée. Le décodeur que nous avons, est simple de part sa construction uniquement 4 couches mais nous obtenons de bons résultats. Cela renforce notre idée de rechercher de nouvelles ressources inexploitées au lieu de s'appuyer sur des ressources traditionnellement utilisées telles que Wikipedia. Dans le but de corriger les différentes erreurs que nous avons listées, des mécanismes d'attention pourraient être appliqués à plusieurs niveaux du système : en entrée de l'encodeur appliqué sur des images de ligne afin de se concentrer sur un mot à la fois, et générer un mot à la fois avec le décodeur ; dans le décodeur pour pouvoir prendre en compte les n-grammes qui ont pu être utilisé à chaque instant $t-1$ pour générer le nouveau caractère. Ces expériences ont été menées uniquement sur des mots et sans ponctuation. La prochaine étape sera donc d'évaluer ce décodeur sur des séquences beaucoup plus longues comme des lignes de titre contenant de la ponctuation.

7 Remerciements

Nous souhaiterions remercier les relecteurs pour leurs suggestions.

Références

BENGIO S. & HEIGOLD G. (2014). Word embeddings for speech recognition. In *Proceedings of the 15th Annual Conference of the International Speech Communication Association (Interspeech'14)*, Singapore.

BLUCHE T., LOURADOUR J. & MESSINA R. (2017). Scan, Attend and Read : End-to-End Handwritten Paragraph Recognition with MDLSTM Attention. In *Proceedings of the 14th International Conference on Document Analysis and Recognition (ICDAR'17)*, Kyoto, Japan.

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2017). Enriching word vectors with subword information. *Transactions of the Association of Computational Linguistics*, **5**(1), 135–146.

BOLLMANN M., BINGEL J. & SØGAARD A. (2017). Learning attention for historical text normalization by learning to pronounce. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL'17)*, p. 332–344, Vancouver, Canada.

CHO K., VAN MERRIENBOER B., BAHDANAU D. & BENGIO Y. (2014a). On the Properties of Neural Machine Translation : Encoder–Decoder Approaches. In *Proceedings of the 8th Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST'14)*, p. 103–111, Doha, Qatar.

CHO K., VAN MERRIËNBOER B., GULCEHRE C., BAHDANAU D., BOUGARES F., SCHWENK H. & BENGIO Y. (2014b). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv :1406.1078*.

CLOPPET F., EGLIN V., KIEU V. C., STUTZMANN D. & VINCENT N. (2016). ICFHR2016 Competition on Classification of Medieval Handwritings in Latin Script. In *Proceedings of the 15th International Conference on Frontiers in Handwriting Recognition (ICFHR'16)*, p. 590–595, Shenzhen, China.

FISCHER A., KELLER A., FRINKEN V. & BUNKE H. (2012). Lexicon-free handwritten word spotting using character HMMs. *PRL*, **33**(7), 934–942.

GARRETTE D. & ALPERT-ABRAMS H. (2016). An unsupervised model of orthographic variation for historical document transcription. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL-HTL'16)*, p. 467–472, San Diego, CA, USA.

GRANELL E., CHAMMAS E., LIKFORMAN-SULEM L., MARTÍNEZ-HINAREJOS C.-D., MOKBEL C. & CÎRSTEA B.-I. (2018). Transcription of spanish historical handwritten documents with deep neural networks. *Journal of Imaging*, **4**(1), 15.

GROSICKI E. & EL-ABED H. (2011). ICDAR 2011 - French Handwriting Recognition Competition. In *Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR'11)*, p. 1459–1463, Beijing, China.

LLADÓS J., RUSIÑOL M., FORNÉS A., FERNÁNDEZ D. & DUTTA A. (2012). On the influence of word representations for handwritten word spotting in historical documents. *IJPRAI*, **26**(05), 1263002–1–25.

NAIR V. & HINTON G. E. (2010). Rectified Linear Units Improve Restricted Boltzmann Machines. In *Proceedings of the 27th international conference on machine learning (ICML'10)*, p. 807–814, Haifa, Israel.

NAKAYAMA H. & NISHIDA N. (2017). Zero-resource machine translation by multimodal encoder–decoder network with multimedia pivot. *Machine Translation*, **31**(1-2), 49–64.

PAN S. J. & YANG Q. (2010). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, **22**(10), 1345–1359.

PRATIKAKIS I., ZAGORIS K., BARLAS G. & GATOS B. (2016). ICFHR2016 Handwritten Document Image Binarization Contest (H-DIBCO 2016). In *Proceedings of the 15th International Conference on Frontiers in Handwriting Recognition (ICFHR'16)*, p. 619–623, Shenzhen, China.

ROMERO V., FORNÉS A., SERRANO N., SÁNCHEZ J. A., TOSELLI A. H., FRINKEN V., VIDAL E. & LLADÓS J. (2013). The ESPOSALLES database : An ancient marriage license corpus for off-line hwr. *PR*, **46**(6), 1658–1669.

SANCHEZ J. A., ROMERO V., TOSELLI A. H., VILLEGAS M. & VIDAL E. (2017). ICDAR2017 Competition on Handwritten Text Recognition on the READ Dataset. In *Proceedings of the 14th International Conference on Document Analysis and Recognition (ICDAR'17)*, p. 1383–1388, Kyoto, Japan.

VANIA C. & LOPEZ A. (2017). From Characters to Words to in Between : Do We Capture Morphology ? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL'17)*, p. 2016–2027, Vancouver, Canada.

VINYALS O., TOSHEV A., BENGIO S. & ERHAN D. (2015). Show and tell : A neural image caption generator. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, p. 3156–3164 : IEEE.