XXXIIe Journées d'Études sur la Parole
4-8 June 2018, Aix-en-Provence, France

# Etude de performance des réseaux neuronaux récurrents dans le cadre de la campagne d'évaluation Multi-Genre Broadcast challenge 3 (MGB3)

Salima Mdhaffar Antoine Laurent Yannick Estève
Laboratoire d'Informatique de l'Université du Mans (LIUM), Avenue Laennec, Le Mans, France
prenom.nom@univ-lemans.fr

RÉSUMÉ

Ces dernières années, l'utilisation des réseaux neuronaux est devenue incontournable dans de nombreux domaines et notamment en traitement automatique des langues. Le travail présenté dans cet article s'inscrit dans le cadre de leur utilisation dans le domaine de la reconnaissance automatique de la parole. Nous présentons les résultats obtenus par des réseaux neuronaux récurrents (RNN) de natures différentes (LSTM, GRU, GRU-Highway) sur les données de la campagne d'évaluation MGB 3. Les données de cette campagne, qui n'est pas encore terminée, correspondent à des enregistrements d'émissions très diverses de la chaîne de télévision britannique BBC. Nos expériences offrent une comparaison des résultats des différents RNN et comment, en combinant des réseaux de neurones récurrents et des modèles de langage N-gram classiques modélisant les phrases dans les deux sens de lecture, il est possible d'améliorer de manière très significative les performances d'un système de reconnaissance de la parole.

ABSTRACT

**Studying performances of recurrent neural networks in the context of the Multi-Genre Broadcast challenge 3 (MGB3) evaluation campaign**

In recent years, the use of neural networks has become indispensable in many fields related to automatic language processing. The work presented in this paper explores the use of several recurrent neural network (RNN) variants (LSTM, GRU and GRU-Highway) in automatic speech recognition in the context of MGB 3 evaluation campaign. The data for this campaign, which is still undergoing, corresponds to a wide variety of emissions recorded from the British television channel BBC. Besides comparing the impact of different RNN, we also show how a combination of RNN and N-gram language models processing the sentences in both reading directions, significantly improves the performance of a speech recognition system.

MOTS-CLÉS : Reconnaissance Automatique de la Parole, Modèle de Langage, Réseaux de Neurones Récurrents, Modèle de Langage à l'Arrière, Interpolation.

KEYWORDS: Automatic Speech Recognition, Language Model, Recurrent Neural Networks, Reverse Language Model, Interpolation.



10.21437/JEP.2018-20

# 1 Introduction

De nos jours, la reconnaissance automatique de la parole est un domaine très actif et a connu une évolution technologique et scientifique très rapide. La raison primordiale de cette évolution est l'utilisation des réseaux de neurones dans la modélisation acoustique et linguistique.

Un modèle de langage (ML) constitue un composant très important dans un système de reconnaissance de la parole. Un ML a pour but de guider le décodeur à choisir la séquence de mots la plus probable. Depuis longtemps, les modèles n-gram sont les plus employés dans la modélisation statistique des langues du fait de leur mise en œuvre rapide et de leur robustesse. Un modèle de langage n-gram estime la probabilité d'apparition d'un mot sachant les n-1 mots qui le précèdent $P(m_i|m_{i-n+1}, ..., m_{i-2}, m_{i-1})$.

Cependant, les modèles N-gram présentent certaines limites. Ils modélisent mal les contraintes à longue distance et nécessitent l'utilisation de techniques de lissage (Chen & Goodman, 1996) pour pallier le problème des événements non vus dans le corpus d'apprentissage. Les techniques les plus connues, sont les techniques de décomptes de Good-Turing (Good, 1953), de Witten-Bell (Witten & Bell, 1991) et de Kneser-Ney (Kneser & Ney, 1995) qui utilisent toute une stratégie de repli (*back-off*).

Malgré la multitude des méthodes de lissage développées, les modèles n-gram prennent mal en compte le fait que les mots dont le sens ou la morphosyntaxe sont proches peuvent s'apparaître dans des contextes similaires. Ceci est dû à la représentation des mots dans un espace discret (le vocabulaire) où il n'existe aucun partage d'information morphologique, syntaxique ou sémantique entre les mots.

Les modèles n-gram basés sur les classes (Brown *et al.*, 1992) ont été introduits afin d'aborder ce problème en regroupant les mots et les contextes dans des classes en fonction de leurs utilisations. L'exploitation des informations relatives aux classes permet d'améliorer la généralisation d'estimation des modèles de langage. Cependant, les problèmes auxquels sont confrontés ces modèles de type n-classes sont nombreux. Le premier problème est que ce type de modèle exige d'avoir un corpus d'apprentissage pré-étiqueté. L'étiquetage manuel est une tâche très coûteuse. En plus, ces modèles possèdent aussi le même inconvénient que celui des modèles N-grams en ce qui concerne la taille de l'historique pris en compte. En effet, la majorité des modèles présentés dans la littérature se limitent à un historique restreint à 3 ou 4 classes (ou mots).

Ces dernières décennies, l'utilisation des réseaux de neurones dans la modélisation du langage a connu beaucoup de succès et a permis l'obtention de performances très intéressantes. Le principe de base de ces modèles s'appuie sur la projection des mots du contexte dans un espace continu ce qui permet d'exploiter la notion de similarité entre les mots. La force des modèles de langage neuronaux réside dans leur capacité de bien généraliser les N-grams non vus car les mots similaires vont avoir probablement le même contexte. Plusieurs travaux dans la littérature prouvent que les modèles de langage neuronaux donnent des résultats plus performants que les modèles de langage n-gram (Mikolov *et al.*, 2011; Schwenk & Gauvain, 2002; Schwenk, 2007).

Dans cet article, nous étudions l'utilisation des modèles de langage neuronaux récurrents dans le cadre de la tâche de reconnaissance automatique de la parole. Nous explorons également l'interpolation de plusieurs types de modèles récurrents et de modèles n-gram modélisant les phrases dans les deux sens de lecture. Ces travaux ont été réalisé dans le cadre de la participation du LIUM dans la campagne d'évaluation MGB.

La suite de l'article est organisée comme suit. Les réseaux de neurones récurrents sont détaillés dans la section 2. Dans la section 3, nous décrivons en détail les implémentations effectuées. La section 4 est consacrée à la présentation du corpus sur lequel nous évaluons les modèles, et présente les différents résultats obtenus. La section 5 conclut l'article.

# 2 Etat de l'art sur les réseaux de neurones récurrents (RNN)

Plusieurs variantes de réseaux de neurones récurrents existent dans la littérature. Les modèles de langage basés sur les réseaux de neurones récurrents sont composés d'au minimum trois couches, à savoir une couche d'entrée, une ou plusieurs couches cachées et une couche de sortie. L'entrée du réseau à l'instant $t$ est $x(t)$, la sortie est notée $y(t)$, et $h(t)$ est la couche cachée. La couche d'entrée (équation 1) est formée d'un vecteur $w(t)$ qui contient la représentation continue du mot actuel et d'un vecteur $s(t-1)$ qui représente les valeurs de sortie dans la couche cachée à partir de l'étape précédente.

$$x(t) = w(t) + s(t-1). \tag{1}$$

Les vecteurs $w(t)$ et $s(t-1)$ sont concaténés dans un seul vecteur afin de former l'entrée de la couche cachée $s(t)$. La fonction d'activation $g$ (équation 2) de la couche cachée est une fonction non linéaire (généralement une sigmoïde (Han & Moraga, 1995)).

$$h_t = g(w(t), s(t-1)) \tag{2}$$

La couche de sortie $y_t$ (équation 3) est constituée d'un nombre de neurones qui est égal à la taille du vocabulaire $v$ ou à la taille de la *shortlist* [1]. Le but de la couche de sortie est de fournir les probabilités de chaque mot w dans le vocabulaire ou de la *shortlist* en fonction de l'historique. La couche de sortie utilise la fonction d'activation softmax (Gao & Pavel, 2017) afin de garantir que la somme des probabilités est égale à 1.

$$y_t = softmax(v * h_t + b) \tag{3}$$

où $b$ est un vecteur de biais.

Bien que les RNN puissent, en théorie, modéliser des dépendances infiniment longues, ils ne sont pas capables de mémoriser des historiques de grande taille. Les réseaux de neurones utilisant des unités de type *Long Short-Term Memory* (LSTM) sont des variantes des réseaux de neurones récurrents dont les unités de base intégrent différentes portes (Hochreiter & Schmidhuber, 1997) (en anglais *gate*), permettant d'écrire, de mettre à jour ou de lire une mémoire contextuelle, à partir d'informations vues précédemment. Ces portes permettent aux LSTMs de modéliser plus efficacement les dépendances longue distance.

Un LSTM est composé d'une mémoire et de trois portes. La porte d'oubli $f$ (forget) contrôle quelle est la partie de la cellule précédente qui sera oubliée. La porte d'entrée $i$ (input) doit choisir les informations pertinentes qui seront transmises à la mémoire. La sortie $o$ (output) contrôle quelle partie de l'état de la cellule sera exposée en tant qu'état caché.

1. la *shortlist* est généralement une liste contenant les mots les plus fréquents dans le corpus d'apprentissage (Schwenk, 2007)

Une autre variation des LSTMs sont les *Gated Recurrent Unit* (GRU) (Cho *et al.*, 2014) qui sont plus simples que le LSTM. Ils ont l'avantage d'être moins coûteux en calculs car ils possèdent moins de paramètres. Ils incorporent seulement deux types de portes au lieu de trois : une porte de réinitialisation $r$ (reset) qui détermine comment combiner la nouvelle entrée avec la mémoire précédente et une porte de modification $u$ (update) qui permet de décider si l'état caché $h$ doit être mis à jour avec le nouvel état caché $h$ ou non.

Une autre extension du GRU est le GRU-Highway. Le réseau highway (Srivastava *et al.*, 2015) est une approche proposée pour optimiser les réseaux et augmenter leurs profondeurs. Le GRU-Highway sert à calculer une sortie qui est une combinaison entre l'entrée et le résultat du GRU. On a $h_t^{(gru)}$ le résultat du GRU classique, $x_t$ est l'entrée à l'instant t et $g$ est une fonction sigmoïde. Les équations pour le GRU-Highway sont les suivantes :

$$g_t = g(x(t), s(t-1)) \tag{4}$$

$$h_t = g_t.h_t^{(gru)} + (1 - g_t).x_t \tag{5}$$

Dans cette étude, nous nous intéressons aux réseaux de neurones récurrents dans le cadre de la modélisation de langage pour la reconnaissance de la parole (RAP).

# 3 Implémentation

Afin de comparer les performances de différents RNNs, plusieurs implémentations ont été réalisées. Cette section détaille ces implémentations.

## 3.1 Implémentation d'un modèle de langue N-gram

Un modèle de langage 3-gram a été construit comme système de base afin de pouvoir mesurer l'amélioration en terme de taux d'erreur de mots (**W**ord **E**rror **R**ate : WER) apportée par les différents modèles neuronaux. Ainsi, ce modèle trigram va être utilisé pour le décodage de la parole afin de construire une liste de N meilleures hypothèses (la liste de N-best). Ce modèle trigram va servir aussi par la suite pour effectuer une interpolation linéaire avec les modèles de langage neuronaux. Par interpolation d'un modèle de langage n-gram avec un modèle de langage RNN, nous entendons une combinaison linéaire de probabilités des phrases (N-best) obtenues à partir de différents modèles, avec des coefficients de pondération pour chaque modèle.

Pour l'interpolation, nous avons aussi construit un modèle 4-gram pour le comparer avec un modèle 3-gram.

## 3.2 Implémentation de trois modèles RNN : GRU, LSTM et GRU-Highway

Trois modèles récurrents : GRU, LSTM et GRU-Highway sont implémentés en utilisant l'outil CUED-RNNLM [2] (Chen *et al.*, 2016). CUED-RNNLM est un outil libre destiné à la modélisation

2. http ://mi.eng.cam.ac.uk/projects/cued-rnnlm/

du langage avec les réseaux de neurones. Il comprend plusieurs types de réseaux de neurones tels que le modèle de langage neuronal "Feed forward" et plusieurs variétés de RNN. La boîte à outils CUED-RNNLM fournit aussi des recettes pour diverses fonctions, notamment l'évaluation de la perplexité, le ré-évaluation de N-best, etc.

Pour garantir une comparaison équitable entre tous les modèles, nous avons utilisé les mêmes réglages et paramètres pour tous les modèles RNN. Le nombre de couches cachées est 2. La taille de la couche cachée est de 200. La taille du vocabulaire de sortie est de 30000 mots. Un modèle avec 60000 mots a également été implémenté pour évaluer l'apport de l'augmentation de la taille du vocabulaire.

### 3.3 Implémentation d'un RNN arrière

Généralement, un modèle de langage est entraîné dans une seule direction : du passé au futur. Cependant, même les données du futur peuvent aider un modèle à estimer la probabilité d'apparition d'un mot. Alors, il est avantageux de construire un modèle de langage dans lequel l'ordre des mots est inversé. Un modèle de langage dont l'ordre des mots est inversé estime la probabilité d'un mot sachant le contexte futur $P(w_\alpha|w_{\alpha+1}, ..., w_{\alpha+n})$ où $\alpha$ est l'indice du mot courant et n le nombre de mots dans le contexte. Le RNN arrière que nous avons implémenté, est similaire à un RNN avant à l'exception que pendant l'apprentissage du modèle la phrase est donnée à l'envers : le premier mot devient le dernier et le dernier mot devient le premier.

Le but d'implémenter un RNN arrière est d'être capable de prendre en compte l'information passée et future mémorisée par interpolation des deux modèles arrière et avant pour effectuer les estimations.

Par interpolation d'un modèle de langage RNN arrière avec un modèle de langage RNN avant, nous entendons une combinaison linéaire de probabilités des phrases (N-best) obtenues à partir du modèle RNN avant et à partir du modèle RNN arrière.

Un modèle 3-gram et un modèle 4-gram arrières ont également été entrainés selon le même principe dans le but d'interpoler les quatre modèles : modèle N-gram avant, modèle N-gram arrière, modèle RNN avant, modèle RNN arrière.

## 4 Résultats expérimentaux

### 4.1 Description du système de RAP

Le système de RAP utilisé pour les expériences présentées dans ce papier est un système préliminaire développé dans le cadre de la campagne d'évaluation MGB 2017. Un premier système de type HMM-GMM contexte dépendant (3-phones), MFCC+LDA+MLLT a été entrainé pour générer les alignements du corpus. Une technique de perturbation de la vitesse (Ko *et al.*, 2015) a été utilisée pour multiplier par 3 la quantité des données d'entrainement.

Ensuite, un modèle de type chain-TDNN (Lattice-free MMI TDNN (Povey *et al.*, 2016)) avec un entraiement discriminant visant à minimiser le risque bayésien sur les états (sMBR - (Kingsbury *et al.*, 2012)) a été entrainé. Des iVecteurs ont également été mis en oeuvre pour l'adaptation instantanée des réseaux de neurones (Saon *et al.*, 2013).

Les phonétisations des mots sont obtenues à l'aide d'un lexique réalisé manuellement, dérivé de Combilex, fourni par les organisateurs de la campagne d'évaluation.

## 4.2 Corpus

Afin de valider notre implémentation du réseau de neurones et comparer nos résultats, nous avons mené des expériences sur les données de la compagne d'évaluation MGB 2017 [3] (**M**ulti **G**enre **B**roadcast) pour la tâche en anglais. MGB est une compagne d'évaluation pour la transcription automatique des émissions TV.

Les données de la compagne d'évaluation MGB 2017 comprennent environ 328 heures d'audio enregistrées sur sept semaines à partir de toutes les chaînes de télévision de BBC (BBC1, BBC2, BBC3, BBC4, etc). Les données couvrent une grande variété de genres (documentaires, actualités, drames ,etc). Quelques statistiques du corpus sont présentées dans le tableau 1.

La campagne MGB n'étant pas encore terminée, les données de test ne sont pas encore fournies. Par conséquent, nous avons éliminé les données de développement du corpus d'apprentissage et nous les avons utilisées pour évaluer notre système.

| Corpus | Corpus d'apprentissage | Corpus de développement |
|---|---|---|
| # segments | 237068 | 5856 |
| # locuteurs | 2719 | 302 |
| Durée | 324h | 4h37 |

TABLE 1 – Statistiques du corpus MGB3

Ainsi, pour l'apprentissage des modèles de langage, nous avons utilisé les données fournies par la compagne d'évaluation MGB 2017. En effet, la compagne d'évaluation MGB fournit des données pour la modélisation acoustique ainsi que des données pour la modélisation linguistique [4]. Quelques statistiques des données d'apprentissage des modèles de langage sont présentées dans le tableau 2.

| | Corpus d'apprentissage |
|---|---|
| # mots total | 645758382 |
| Vocabulaire | 757748 |
| Vocabulaire utilisé pour les ML n-gram | 164000 |

TABLE 2 – Statistiques du corpus d'apprentissage pour la modélisation linguistique

## 4.3 Analyse des résultats

La qualité des modèles de langage implémentés est évaluée en terme de gain en WER. WER (Pallett, 2003) est la métrique d'évaluation couramment utilisée dans la littérature pour l'analyse des performances d'un système de reconnaissance de la parole. Elle se calcule comme suit :

$$WER = \frac{S + I + D}{N} \tag{6}$$

3. http ://www.mgb-challenge.org/
4. http ://www.mgb-challenge.org/download.html

Où S est le nombre de mots remplacés par le système, I est le nombre de mots insérés par le système, D est le nombre de mots supprimés par le système et N est le nombre total de mots dans une phrase.

Le tableau 3 présente les différents résultats expérimentaux. Afin d'obtenir ces résultats, deux passes sont effectués : la première passe consiste à obtenir une liste de N-best (les N meilleures hypothèses) avec N=200 en utilisant le système de reconnaissance décrit, la deuxième passe consiste à attribuer des scores à ces N-bests en utilisant les modèles de langage implémentés. La troisième colonne présente les résultats en terme de WER. La colonne 4 ($\delta$) présente le gain absolu par rapport au système de base sans effectuer la deuxième passe (1-best). Les poids d'interpolation utilisés sont 0,5 dans le cas d'interpolation de deux modèles et 0,25 dans le cas d'interpolation de quatre modèles.

| | | WER % | $\delta$ |
|---|---|---|---|
| 1 | 3-gram (système de base) | 24,7 | - |
| 2 | LSTM | 22,6 | 2,1 |
| 3 | GRU | 22,5 | 2,2 |
| 4 | GRU-Highway | 22,3 | 2,4 |
| 5 | LSTM + 3-gram | 22,1 | 2,6 |
| 6 | GRU + 3-gram | 22,0 | 2,7 |
| 7 | GRU + 4-gram | 21,7 | 3 |
| 8 | GRU-Highway + 3-gram | 21,6 | 3,1 |
| 9 | GRU + 3-gram (\|shortlist\|= 60K) | 21,8 | 2,9 |
| 10 | GRU + 4-gram (\|shortlist\|= 60K) | 21,6 | 3,1 |
| 11 | GRU arrière | 22,5 | 2,2 |
| 12 | GRU arrière + 3-gram arrière | 22,2 | 2,5 |
| 13 | GRU arrière + 3-gram arrière + GRU avant + 3-gram avant | 21.6 | 3,1 |
| 14 | GRU arrière + 4-gram arrière + GRU avant + 4-gram avant | 21.4 | 3,3 |

TABLE 3 – Résultats obtenus

Les résultats obtenus par les différents types de réseaux de neurones sont meilleurs que ceux obtenus avec le système de base (1) : decodé avec un modèle n-gram sans passer par la deuxième passe du rescoring.

Les résultats expérimentaux présentés dans le tableau 3 montrent que le GRU-Highway (4) a donné un résultat plus performant que le LSTM (2) et le GRU (3) en terme de WER (23,3% WER pour le GRU-Highway, 22,6 %WER pour le LSTM, 22,5 %WER pour le GRU).

L'interpolation de RNN avec un modèle n-gram permet une réduction 0,5% absolu de WER dans le cas de GRU+3-gram (6) par rapport au GRU (3).

En plus, l'interpolation avec un modèle 4-gram est plus performante que l'interpolation avec un modèle 3-gram. Le résultat d'interpolation d'un modèle GRU avec un 4-gram (7) est 21,7% WER par contre le résultat de l'interpolation d'un modèle GRU avec un 3-gram (6) est 22% WER.

Les résultats obtenus montrent que notre proposition de combiner un RNN arrière avec un RNN avant est utile et améliore significativement les résultats en termes de WER (13,14). Ceci confirme l'utilité de l'utilisation des informations du contexte passé et futur à la fois.

Enfin, nous avons montré que l'utilisation de 60000 mots dans la shortlist (9,10) a donné de meilleurs résultats par rapport à l'utilisation de 30000 mots dans shortlist (6,7).

# 5 Conclusion

Le travail présenté dans cet article est une étude portant sur l'utilisation des réseaux neuronaux pour la modélisation du langage dans le cadre de la reconnaissance automatique de la parole. Nous nous sommes intéressés en particulier aux réseaux neuronaux récurrents. Plusieurs types de réseaux neuronaux récurrents ont été évalués et expérimentés. Ainsi, nous avons exploré dans ce travail l'interpolation des modèles de langage neuronaux avec des modèles de langage n-gram ainsi que l'interpolation avec un modèle de langage récurrent arrière. Les expériences sont effectuées dans le cadre de la campagne d'évaluation MGB 2017. Les résultats obtenus montrent que les réseaux de neurones récurrents ainsi que les deux types d'interpolations apportent des améliorations significatives en terme de taux d'erreur mots.

# Remerciements

Nous remercions l'agence ANR pour son financement à travers le projet PASTEL sous le numéro de contrat ANR-16-CE33-0007.

# Références

BROWN P. F., DESOUZA P. V., MERCER R. L., PIETRA V. J. D. & LAI J. C. (1992). Class-based n-gram models of natural language. *Computational linguistics*, **18**(4), 467–479.

CHEN S. F. & GOODMAN J. (1996). An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, p. 310–318 : Association for Computational Linguistics.

CHEN X., LIU X., QIAN Y., GALES M. & WOODLAND P. C. (2016). Cued-rnnlm—an open-source toolkit for efficient training and evaluation of recurrent neural network language models. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, p. 6000–6004 : IEEE.

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

GAO B. & PAVEL L. (2017). On the properties of the softmax function with application in game theory and reinforcement learning. *arXiv preprint arXiv :1704.00805*.

GOOD I. J. (1953). The population frequencies of species and the estimation of population parameters. *Biometrika*, **40**(3-4), 237–264.

HAN J. & MORAGA C. (1995). The influence of the sigmoid function parameters on the speed of backpropagation learning. *From Natural to Artificial Neural Computation*, p. 195–201.

HOCHREITER S. & SCHMIDHUBER J. (1997). Long short-term memory. *Neural computation*, **9**(8), 1735–1780.

KINGSBURY B., SAINATH T. N. & SOLTAU H. (2012). Scalable minimum bayes risk training of deep neural network acoustic models using distributed hessian-free optimization. In *Thirteenth Annual Conference of the International Speech Communication Association*.

KNESER R. & NEY H. (1995). Improved backing-off for m-gram language modeling. In *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, volume 1, p. 181–184 : IEEE.

KO T., PEDDINTI V., POVEY D. & KHUDANPUR S. (2015). Audio augmentation for speech recognition. In *Sixteenth Annual Conference of the International Speech Communication Association*.

MIKOLOV T., KOMBRINK S., BURGET L., ČERNOCKỲ J. & KHUDANPUR S. (2011). Extensions of recurrent neural network language model. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, p. 5528–5531 : IEEE.

PALLETT D. S. (2003). A look at nist's benchmark asr tests : past, present, and future. In *Automatic Speech Recognition and Understanding, 2003. ASRU'03. 2003 IEEE Workshop on*, p. 483–488 : IEEE.

POVEY D., PEDDINTI V., GALVEZ D., GHAHREMANI P., MANOHAR V., NA X., WANG Y. & KHUDANPUR S. (2016). Purely sequence-trained neural networks for asr based on lattice-free mmi. In *Interspeech*, p. 2751–2755.

SAON G., SOLTAU H., NAHAMOO D. & PICHENY M. (2013). Speaker adaptation of neural network acoustic models using i-vectors. In *ASRU*, p. 55–59.

SCHWENK H. (2007). Continuous space language models. *Computer Speech & Language*, **21**(3), 492–518.

SCHWENK H. & GAUVAIN J.-L. (2002). Connectionist language modeling for large vocabulary continuous speech recognition. In *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, volume 1, p. I–765 : IEEE.

SRIVASTAVA R. K., GREFF K. & SCHMIDHUBER J. (2015). Highway networks. *arXiv preprint arXiv :1505.00387*.

WITTEN I. H. & BELL T. C. (1991). The zero-frequency problem : Estimating the probabilities of novel events in adaptive text compression. *Ieee transactions on information theory*, **37**(4), 1085–1094.