Des Réseaux de Neurones avec Mécanisme d'Attention pour la Compréhension de la Parole

Edwin Simonnet, Paul Deléglise, Nathalie Camelin, Yannick Estève

Des Réseaux de Neurones avec Mécanisme d'Attention pour la Compréhension de la Parole *

Edwin Simonnet Paul Deléglise Nathalie Camelin Yannick Estève
LIUM, Institut d'Informatique Claude Chappe Université du Maine Avenue Laennec, 72085 Le Mans, France
firstname.lastname@univ-lemans.fr

RÉSUMÉ

L'étude porte sur l'apport d'un réseau de neurones récurrent (Recurrent Neural Network - RNN) bidirectionnel encodeur/décodeur avec mécanisme d'attention pour une tâche de compréhension de la parole. Les premières expériences faites sur le corpus ATIS confirment la qualité du système RNN état de l'art utilisé pour cet article, en comparant les résultats obtenus à ceux récemment publiés dans la littérature. Des expériences supplémentaires montrent que les RNNs avec mécanisme d'attention obtiennent de meilleures performances que les RNNs récemment proposés pour la tâche d'étiquetage en concepts sémantiques. Sur le corpus MEDIA, un corpus français état de l'art pour la compréhension dédié à la réservation d'hôtel et aux informations touristiques, les expériences montrent qu'un RNN bidirectionnel atteint une f-mesure de 79,51 tandis que le même système intégrant le mécanisme d'attention permet d'atteindre une f-mesure de 80,27.

ABSTRACT

Exploring the use of Attention-Based Recurrent Neural Networks For Spoken Language Understanding

This study explores the use of a bidirectional recurrent neural network (RNN) encoder/decoder based on a mechanism of attention for a Spoken Language Understanding (SLU) task. First experiments carried on the ATIS corpus confirm the quality of the RNN baseline system used in this paper, by comparing its results on the ATIS corpus to the results recently published in the literature. Additional experiments show that RNN based on a mechanism of attention performs better than RNN architectures recently proposed for a slot filling task. On the French MEDIA corpus, a French state-of-the-art corpus for SLU dedicated to hotel reservation and tourist information, experiments show that a bidirectionnal RNN reaches a f-measure value of 79.51 while the use of a mechanism of attention allows us to reach a f-measure value of 80.27.

MOTS-CLÉS : Compréhension de la Parole, Réseaux de Neurones Récurrents, Mécanisme d'Attention, Bidirectionnel.

KEYWORDS: Spoken Language Understanding, Recurrent Neural Networks, Attention Based Mechanism, Bidirectional.

*. Cet article présente le travail d'un commencement de thèse. Il est traduit de l'article publié en anglais à la conférence NIPS 2015 (Simonnet *et al.*, 2015)

1 Introduction

La compréhension de la parole (Spoken Language Understanding – SLU) peut être définie comme l'interprétation des signaux transportés par un signal de parole (De Mori *et al.*, 2008). Cette interprétation est habituellement assimilée à l'extraction et la représentation du *sens* contenu par les mots d'une phrase parlée.

1.1 La tâche d'étiquetage en concepts sémantiques

De nos jours, extraire le sens d'un discours reste une opération très complexe et la SLU, qui en est une application, est souvent réduite à la construction d'une représentation sémantique spécifique à la tâche.

Dans ce contexte la SLU correspond à une tâche d'étiquetage en concepts sémantiques qui est l'extraction d'une séquence de concepts sémantiques à partir d'une séquence de mots donnée (Hahn *et al.*, 2011). Dans le passé, plusieurs méthodes d'étiquetage en concepts sémantiques ont été proposées dans ce cadre. Jusqu'à il y a deux ans, les champs aléatoires conditionnels (Lafferty *et al.*, 2001)(Conditional Random Field – CRF) étaient considérés comme l'approche état de l'art (Hahn *et al.*, 2011).

1.2 Objectif

Récemment, il a été montré dans (Mesnil *et al.*, 2013, 2015) que les réseaux de neurones récurrents (Recurrent Neural Network - RNN) peuvent atteindre de meilleures performances que les CRFs dans une tâche d'étiquetage en concepts sémantiques appliquée au corpus de réservation de vol ATIS (Hemphill *et al.*, 1990). Néanmoins, (Vukotic *et al.*, 2015) a démontré que ces meilleures performances des RNNs ne se renouvellent pas sur un corpus de SLU plus complexe tel que le corpus français MEDIA (Bonneau-Maynard *et al.*, 2009) (Devillers *et al.*, 2004). En effet dans cette dernière étude, les CRFs ont obtenu des résultats significativement meilleurs que ceux des RNNs. Cela peut être expliqué par le fait que la tâche MEDIA semble plus difficile à traiter que celle d'ATIS, on remarque notamment que la taille du vocabulaire et la proportion de mots du corpus étant associé à un concept sont plus grandes dans le corpus MEDIA que dans le corpus ATIS.

Dans cet article, nous ne souhaitons pas établir si les CRFs ou les réseaux de neurones profonds (Deep Neural Network - DNN) constituent l'état de l'art courant pour la SLU. Nous sommes convaincus du potentiel des architectures DNN et nous avons pour objectif d'étudier l'utilisation d'un RNN avec mécanisme d'attention (Graves, 2013) initialement dédié à la reconnaissance d'écriture manuelle et ayant été utilisé avec succès pour la reconnaissance de la parole (Chorowski *et al.*, 2014). Étant donné que la tâche de SLU MEDIA semble plus complexe que celle d'ATIS, nous nous sommes focalisé sur le corpus MEDIA pour évaluer les conséquences de l'utilisation du mécanisme d'attention proposé dans (Bahdanau *et al.*, 2014).

1.3 Les principes généraux d'un RNN avec mécanisme d'attention

De très bonnes descriptions des principes des RNNs avec mécanisme d'attention peuvent être trouvées dans (Bahdanau *et al.*, 2014; Chorowski *et al.*, 2014, 2015). Le mécanisme d'attention fut

intuitivement conçu afin de prendre en compte la position des éléments d'entrée lors de l'encodage d'une séquence dans une approche avec un RNN encodeur-décodeur. Des poids, ré-estimés après chaque génération de sortie, sont attribués aux annotations (correspondants aux mots) en entrée. Cela permet au décodeur de décider les parties de la phrase d'entrée auxquelles prêter attention et au décodeur de ne pas avoir à encoder automatiquement toute l'information. Dans cet article, le RNN avec mécanisme d'attention s'inspire largement de l'architecture proposée dans (Bahdanau *et al.*, 2014) pour la traduction automatique, comme décrit dans la figure 1. Nous souhaitons adapter cette méthode pour la SLU en considérant le processus d'étiquetage en concepts sémantiques similaire à un problème de traduction depuis des mots (langage source) vers des concepts sémantiques (langage cible).

Cette architecture se base sur un RNN bidirectionnel comme encodeur. Ce RNN bidirectionnel calcul une annotation h_i pour chaque mot w_i de la séquence d'entrée $\{w_1, ... ,w_I\}$. Cette annotation est la concaténation des couches cachées correspondantes avant (forward) et arrière (backward) obtenues respectivement par le RNN avant et le RNN arrière constituant le RNN bidirectionnel. Chaque annotation contient le résumé d'à la fois les mots précédents et les mots suivants. Étant donné que les couches cachées des RNNs tendent à mieux représenter les entrées récentes, chaque annotation h_i se concentre sur les mots autour de w_i.

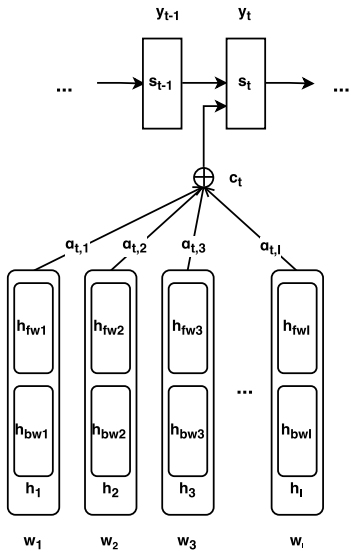

FIGURE 1 – Illustration d'un RNN avec mécanisme d'attention de (Bahdanau *et al.*, 2014)

Après avoir appliqué cet encodeur bidirectionnel, pour chaque mot à l'intérieur de la séquence d'entrée, une annotation est calculée : cette séquence d'annotations $\{h_1, ... ,h_I\}$ sera utilisée par le décodeur pour calculer un vecteur de contexte c_t. Un vecteur de contexte est recalculé après chaque émission d'une étiquette en sortie. Ce calcul prend en compte une somme pondérée de toutes les annotations calculées par l'encodeur. Cette pondération dépend de la cible en sortie courante et constitue le coeur du mécanisme d'attention : une bonne estimation des poids α_{ti} permet au décodeur de choisir les parties de la séquence d'entrée auxquelles il doit prêter attention. Ce vecteur de contexte

sera utilisé par le décodeur conjointement avec l'étiquette émise en sortie précédemment y_{t-1} et l'état courant s_t de la couche cachée du RNN afin de prendre une décision concernant l'étiquette en sortie courante y_t.

2 Implémentation

Afin de comparer les RNNs avec et sans mécanisme d'attention pour une tâche SLU, des implémentations fournies par les auteurs de (Mesnil *et al.*, 2013) et (Bahdanau *et al.*, 2014) ont été utilisées pour nos expériences. L'implémentation RNN venant de (Mesnil *et al.*, 2013) ne correspond pas à celle utilisée pour leurs expériences : seulement l'implémentation du RNN avant est disponible tandis que cette étude utilise un réseau de neurone bidirectionnel. Afin de valider notre implémentation de ce réseau de neurones bidirectionnel, des premières expériences ont été faites sur le corpus ATIS afin de comparer nos résultats avec ceux présentés dans (Mesnil *et al.*, 2013).

2.1 Implémentation d'un RNN

Notre implémentation d'un RNN se base sur (Mesnil *et al.*, 2013) et plus précisément sur l'implémentation proposée par son premier auteur (Mesnil, 2015) d'un RNN avant de type elman/jordan avec le paramètre T fixé à 1. Ce paramètre indique la récupération des T étapes temporelles précédentes depuis la couche de sortie dans un réseau de type jordan ou depuis la couche cachée dans un réseau de type elman. À partir de cette implémentation, nous utilisons un RNN de type elman [1] dont les couches cachées et couches de sorties sont calculées comme suit :

$$couche\ cachee : h(t) = sigmoid(W_x.x(t) + W_h.h(t-1) + b_h)) \ (1)$$

$$couche\ de\ sortie : s(t) = softmax(W.h(t) + b)) \ (2)$$

où $x(t)$ est le mot d'entrée du RNN (représentation continue de mot - embedding) à t et $h(t-1)$ la sortie de la couche cachée à $t-1$. Les paramètres W_x, W_h et W du RNN sont les matrices de poids, b_h et b les biais, et h_0 la couche cachée initiale de l'étape précédente pour le premier mot de la séquence pour lequel rien n'a encore été calculé. Les paramètres sont ajustés au cours des époques d'apprentissage à l'aide d'une descente de gradient effectuée sur des *mini-batches*.

La version arrière est implémentée à partir de la version avant fournie dans (Mesnil, 2015). Un RNN arrière est similaire à un avant à l'exception que la prédiction est faite du futur vers le passé. La phrase est donnée à l'envers pour simuler cet effet. W_h représente la matrice de poids entre la couche cachée de l'étape prochaine et la courante. h_0 est la couche cachée initiale de l'étape prochaine pour le dernier mot de la phrase (*i.e.* le premier mot donné au RNN). La couche de sortie est toujours calculée à l'aide de l'équation (2). La couche cachée est calculée comme suit :

$$h(t) = sigmoid(W_x.x(t) + W_h.h(t+1) + b_h)) \ (3)$$

Avec un RNN arrière acquis, le bidirectionnel peut être implémenté. Un RNN bidirectionnel effectue des prédictions prenant en compte le passé et le futur. Par conséquent, des RNNs avant et arrière déjà

1. l'opération serait la même pour un jordan excepté que la couche de sortie est réinjectée dans la couche cachée à t+1

entraînés sont utilisés conjointement. Il y a deux matrices de poids W_h : W_h_fw entre la couche cachée de l'étape précédente et la courante ; et W_h_bw entre la couche cachée de l'étape suivante et la courante. Il en va de même pour b_h_fw et b_h_bw. Finalement il n'y a pas de couches cachées initiales h_0 étant donné que ces dernières sont récupérées depuis les RNNs avant et arrière déjà entraînés. La couche cachée est alors calculée comme suit :

$$h(t) = sigmoid(W_x.x(t) + W_h\_fw.h(t-1) + b_h\_fw + W_h\_bw.h(t+1) + b_h\_bw) \quad (4)$$

Notre objectif est également d'implémenter des dépendances à long termes comme décrit dans (Mesnil *et al.*, 2013) en fournissant au réseau la somme des étapes précédentes/suivantes, c'est à dire avec un T supérieur à 1 selon l'équation :

$$h_{bd}(t) = f(W_x.x(t) + \sum_{k=1}^{T}(W_h\_bw_k.h\_bw(t+k) + b_h\_bw) + \sum_{k=1}^{T}(W_h\_fw_k.h\_fw(t-k) + b_h\_fw)) \quad (5)$$

(Mesnil, 2015) donne une implémentation avec T fixé à 1. Différentes valeurs de T ont été testées sur le corpus ATIS pour les RNNs avant et arrière afin de voir si cet ajout de contexte améliore le système, mais le RNN bidirectionnel atteint ses meilleurs résultats avec des RNNs avant et arrière ayant tous deux T=1.

Les RNNs avant et arrière utilisés pour l'entraînement ou la classification du bidirectionnel sont entraînés individuellement auparavant. Différentes manières d'entraîner ces RNNs ont été testées. D'abord le *parallel train*, qui consiste à entraîner les RNNs avant, arrière et ensuite bidirectionnel à chaque époque. Deuxièmement, l'entraînement *get best*, dans lequel l'entraînement du RNN bidirectionnel se base sur les meilleurs paramètres d'à la fois les RNNs avant et arrière déjà entraînés avant. Enfin, l'apprentissage *train best* qui combine les deux approches précédentes : à chaque époque les RNNs avant et arrière sont entraînés comme dans le *parallel train*. Ensuite le RNN bidirectionnel utilise les paramètres des dernières meilleures époques pour le RNN avant et arrière respectivement comme dans le *get best*.

Les expériences ont montré que le meilleur apprentissage est l'approche *parallel train* suivi du *train best* et enfin du *get best*. Cela peut être expliqué par le fait que le RNN bidirectionnel apprend d'avantage avec des RNNs avant et arrière qui ont une plus grande variabilité au cours des époques même s'ils ne donnent pas toujours les meilleurs résultats. L'apprentissage en est par conséquent plus diversifié. En effet l'approche *get best* utilisant des paramètres avant et arrière fixés à partir de leur meilleures époques est celle qui donne les pires résultats.

2.2 Implémentation d'un RNN bidirectionnel avec mécanisme d'attention

L'implémentation d'un RNN avec mécanisme d'attention utilisé dans notre étude est dérivée de celle utilisée dans (Bahdanau *et al.*, 2014) et disponible à `https://github.com/kyunghyuncho/GroundHog`. Cette implémentation a été créée pour une tâche de traduction automatique. Dans cette tâche, les séquences d'entrée et de sortie ont souvent des longueurs différentes. L'approche avec un RNN encodeur-décodeur est particulièrement bien adaptée pour ce cas de figure. Pour la tâche SLU en particulier, il est très important d'obtenir une correspondance entre les mots (entrées) et les concepts sémantiques (sorties) afin de pouvoir extraire la valeur du concept. Afin d'obtenir un

alignement précis, nous avons modifié le processus de décodage du RNN bidirectionnel en imposant que la séquence d'étiquettes en sortie et la séquence de mots en entrée aient la même taille. C'est la seule modification apportée à l'implémentation venant de (Bahdanau *et al.*, 2014).

3 Expériences

Afin de valider notre propre implémentation du réseau de neurones bidirectionnel et comparer nos résultats avec ceux présentés dans (Mesnil *et al.*, 2013), nous avons mené des premières expériences sur le corpus ATIS. Puis, une fois notre implémentation validée, nous comparons les approches RNN avec et sans mécanisme d'attention sur le corpus MEDIA.

3.1 Validation de l'implémentation d'un RNN sur le corpus ATIS

Le corpus utilisé par (Mesnil *et al.*, 2013) est le corpus ATIS (Airline Travel Information System) spécialisé dans les requêtes de réservation de billets d'avion. Il est composé de 4978/893 (apprentissage/test) phrases annotées selon 128 étiquettes sémantiques. Le corpus d'apprentissage est divisé comme suit : 80% pour l'apprentissage 20% pour la validation.

Afin d'aider le classifieur à délimiter les séquences de mots ayant la même étiquette, une méthode courante consiste à rajouter un suffixe $B/I/O$ aux étiquettes sémantiques, respectivement pour le début (*Beginning*), l'intérieur (*Inside*) et l'extérieur (*Outside*) d'une séquence. Seuls les suffixes B et I sont utilisés ici. O est représenté par l'étiquette $NULL$ qui est associée aux mots ne portant aucune information sémantique selon la tâche spécifique.

L'évaluation est faite en calculant la f-mesure qui utilise les métriques de rappel et de précision. On obtient un score prenant en compte la présence ou l'absence d'un concept dans une phrase sans notion de séquentialité.

$$f - mesure = \frac{2(precision.rappel)}{precision + rappel}$$

Dans (Mesnil *et al.*, 2013) précision et rappel sont définis [2] comme suit :

$$rappel = \frac{nombre\ de\ segments\ corrects}{nombre\ de\ segments\ dans\ la\ reference}$$

$$precision = \frac{nombre\ de\ segments\ corrects}{nombre\ de\ segments\ dans\ l'\ hypothese}$$

Un segment de concept est correct s'il commence et finit avec les mêmes mots pour l'hypothèse et la référence. La f-mesure est maximisée sur le corpus de validation durant le processus d'apprentissage.

Le tableau 1 présente les résultats obtenus par (Mesnil *et al.*, 2013) et ceux par notre implémentation en utilisant les hyper-paramètres suivants : nombre d'époques=100 ; fenêtre=5 ; nombre d'unités dans la couche cachée=200 ; dimension d'embedding=50.

2. Calculés à l'aide du script conlleval.pl fourni par (Mesnil, 2015)

Expérience	Architecture	Type	f-mesure
[Mesnil et al. 2013]	Jordan	bidirectional	93,98
RNN baseline	Elman	bidirectional	94,13

TABLE 1 – Comparaison entre les performances du RNN présenté dans (Mesnil *et al.*, 2013) et notre implémentation d'un RNN état de l'art sur le corpus ATIS.

Comme montré dans le tableau 1, notre système RNN bidirectionnel état de l'art atteint des résultats similaires à ceux du RNN bidirectionnel présenté dans (Mesnil *et al.*, 2013). Cela valide notre implémentation et nous permet d'étudier l'utilisation d'un RNN avec mécanisme d'attention pour une tâche SLU d'étiquetage en concepts sémantiques plus complexe.

3.2 Performances d'un RNN avec mécanisme d'attention sur le corpus MEDIA

Le corpus MEDIA (Bonneau-Maynard *et al.*, 2009) est un corpus de dialogue état de l'art français. Il contient 1257 dialogues entre des utilisateurs et un système simulé (protocole Wizard of Oz) dans le domaine de la réservation d'hôtel et des informations touristiques. Seuls les tours de parole des utilisateurs sont utilisés pour l'apprentissage et la classification. Cet ensemble de tours est divisé en trois sous-corpus : l'ensemble APPRENTISSAGE qui contient 17,6k énoncés, l'ensemble DEV qui contient 1,3k énoncés et enfin l'ensemble TEST qui est composé de 3,5k énoncés.

Chaque énoncé a été manuellement transcrit et annoté en se basant sur 74 étiquettes de concept allant de simples réponses (*e.g.* le mot "oui" est associé au concept *reponse*) à des requêtes spécifiques à la tâche (*e.g.* les mots "avec baignoire" sont associés au concept *equipement_chambre*). Une annotation plus riche est disponible dans MEDIA incluant les modes, les spécifieurs et les valeurs mais pour commencer nous choisissons d'évaluer seulement les étiquettes des concepts sémantiques.

Dans MEDIA, le but du dialogue pour l'utilisateur est d'obtenir des informations qui sont stockées dans une base de données. Par conséquent, les noms de rues, de villes ou d'hôtels, les listes d'équipement de chambre, les types de nourriture, *etc.* sont connus. De plus, des mots plus généraux représentant les nombres, les jours, les mois sont également connus. Tous ces mots (spécifiques à la tâche SLU ou généraux) ont été rassemblés dans un lexique sémantique qui permet d'associer un mot à une classe sémantique.

L'énoncé d'un utilisateur est représenté par une séquence de mots et de classes sémantiques. Si elle existe, le mot est substitué par sa classe sémantique. Un exemple est disponible dans le tableau 2. Comme pour ATIS, les suffixes B/I sont associés aux étiquettes de concepts.

Mots	j'aimerais réserver un hôtel pour les **trois premiers** jours de **Mai** à **Marseille**.
Mots+Class. Sem.	j'aimerais réserver un hôtel pour les **UNIT ORDINAL** jours de **MOIS** à **VILLE**.

TABLE 2 – Représentation d'un énoncé utilisateur par mots et par mots+classes sémantiques.

Le tableau 3 montre les performances mesurées en terme de f-mesure des différentes architectures de RNN sur le corpus MEDIA.

Architecture	f-mesure
RNN avant	74,04
RNN arrière	77,42
RNN bidirectionnel	79,51
Mécanisme d'attention	80,27
RNN encodeur-décodeur sans mécanisme d'attention	38,25

TABLE 3 – Résultats sur MEDIA avec classes sémantiques

Comme prévu, les résultats ne sont pas aussi bon que sur le corpus ATIS. L'architecture RNN bidirectionnel obtient de meilleurs résultats en comparaison avec un RNN arrière ou avant. Cela confirme l'utilité d'utiliser des informations du contexte passé et futur ensemble.

De plus, les résultats présentés dans le tableau 3 montrent que l'encodeur RNN bidirectionnel avec mécanisme d'attention est plus performant qu'un RNN bidirectionnel classique.

Enfin, il est montré qu'un encodeur-décodeur RNN obtient de très faibles performances sans mécanisme d'attention lors de la production d'une séquence d'étiquettes en sortie ayant la même longueur que la séquence de mots en entrée.

4 Conclusion

Cette étude a pour but d'examiner l'utilisation d'un RNN bidirectionnel avec mécanisme d'attention pour la tâche SLU d'étiquetage en concepts sémantiques. Nos expériences montrent que cette architecture atteint de meilleurs résultats qu'une approche plus classique avec un RNN bidirectionnel sur un corpus SLU complexe. Cette approche classique d'un RNN bidirectionnel a été introduite et présentée comme l'approche état de l'art pour la SLU il y a 2 ans par (Mesnil *et al.*, 2013) sur le corpus ATIS. Même si (Vukotic *et al.*, 2015) a montré que les CRFs obtiennent toujours de meilleurs résultats que ce RNN bidirectionnel sur des données plus complexes comme le corpus MEDIA, nos résultats montrent que des approches prometteuses comme le mécanisme d'attention peuvent toujours améliorer les RNN pour la SLU.

Remerciements

Nous remercions l'agence ANR pour son financement à travers CHIST-ERA ERA-Net JOKER sous le numéro de contrat ANR-13-CHR2-0003-05.

De plus, les auteurs remercient Sahar Ghannay pour son aide constructive au cours de l'écriture de cet article.

Références

BAHDANAU D., CHO K. & BENGIO Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv :1409.0473*.

BONNEAU-MAYNARD H., QUIGNARD M. & DENIS A. (2009). Media : a semantically annotated corpus of task oriented dialogs in french. *Language Resources and Evaluation*, **43**(4), 329–354.

CHOROWSKI J., BAHDANAU D., CHO K. & BENGIO Y. (2014). End-to-end continuous speech recognition using attention-based recurrent nn : First results. *arXiv preprint arXiv :1412.1602*.

CHOROWSKI J., BAHDANAU D., SERDYUK D., CHO K. & BENGIO Y. (2015). Attention-based models for speech recognition. *arXiv preprint arXiv :1506.07503*.

DE MORI R., BECHET F., HAKKANI-TÜR D., MCTEAR M., RICCARDI G. & TUR G. (2008). Spoken language understanding. *Signal Processing Magazine, IEEE*, **25**(3), 50–58.

DEVILLERS L., MAYNARD H., ROSSET S., PAROUBEK P., MCTAIT K., MOSTEFA D., CHOUKRI K., CHARNAY L., BOUSQUET C., VIGOUROUX N., BÉCHET F., ROMARY L., ANTOINE J., VILLANEAU J., VERGNES M. & GOULIAN J. (2004). The french media/evalda project : the evaluation of the understanding capability of spoken language dialogue systems. In *LREC*.

GRAVES A. (2013). Generating sequences with recurrent neural networks. *arXiv preprint arXiv :1308.0850*.

HAHN S., DINARELLI M., RAYMOND C., LEFEVRE F., LEHNEN P., DE MORI R., MOSCHITTI A., NEY H. & RICCARDI G. (2011). Comparing stochastic approaches to spoken language understanding in multiple languages. *Audio, Speech, and Language Processing, IEEE Transactions on*, **19**(6), 1569–1583.

HEMPHILL C. T., GODFREY J. J. & DODDINGTON G. R. (1990). The atis spoken language systems pilot corpus. In *Proceedings of the DARPA speech and natural language workshop*, p. 96–101.

LAFFERTY J. D., MCCALLUM J. D. & PEREIRA F. C. N. (2001). Conditional random fields : probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conference on Machine Learning (ICML-2001)*, San Francisco, CA, USA.

MESNIL G. (2015). Recurrent Neural Networks with Word Embeddings DeepLearning 0.1 documentation. `http://www.deeplearning.net/tutorial/rnnslu.html#rnnslu`.

MESNIL G., DAUPHIN Y., YAO K., BENGIO Y., DENG L., HAKKANI-TUR D., HE X., HECK L., TUR G., YU D. *et al.* (2015). Using recurrent neural networks for slot filling in spoken language understanding. *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, **23**(3), 530–539.

MESNIL G., HE X., DENG L. & BENGIO Y. (2013). Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In *INTERSPEECH*, p. 3771–3775.

SIMONNET E., DELÉGLISE P., CAMELIN N. & ESTÈVE Y. (2015). Exploring the use of attention-based recurrent neural networks for spoken language understanding. In *NIPS*.

VUKOTIC V., RAYMOND C. & GRAVIER G. (2015). Is it time to switch to word embedding and recurrent neural networks for spoken language understanding ? In *InterSpeech*.