Réseaux neuronaux profonds pour l'étiquetage de séquences

Yoann Dupont[1] Marco Dinarelli[1] Isabelle Tellier[1]
(1) Lattice (UMR 8094), CNRS, ENS Paris, Université Sorbonne Nouvelle,
PSL Research University, USPC, 1 rue Maurice Arnoux, 92120 Montrouge, France
yoa.dupont@gmail.com, marco.dinarelli@ens.fr,
isabelle.tellier@univ-paris3.fr

RÉSUMÉ

Depuis quelques années les réseaux neuronaux se montrent très efficaces dans toutes les tâches de Traitement Automatique des Langues (TAL). Récemment, une variante de réseau neuronal particulièrement adapté à l'étiquetage de séquences textuelles a été proposée, utilisant des représentations distributionnelles des étiquettes. Dans cet article, nous reprenons cette variante et nous l'améliorons avec une version profonde. Dans cette version, différentes couches cachées permettent de prendre en compte séparément les différents types d'informations données en entrée au réseau. Nous évaluons notre modèle sur les mêmes tâches que la première version de réseau de laquelle nous nous sommes inspirés. Les résultats montrent que notre variante de réseau neuronal est plus efficace que les autres, mais aussi qu'elle est plus efficace que tous les autres modèles évalués sur ces tâches, obtenant l'état-de-l'art.

ABSTRACT

Deep Neural Networks for Sequence Labeling

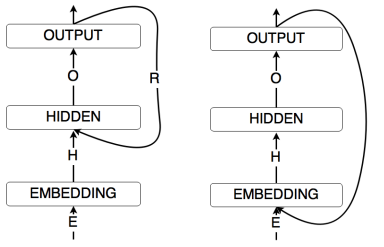
Since a couple of years, neural networks prove to be very effective on all NLP tasks. Recently, a variant of neural network particularly suited for sequence labeling has been proposed, which uses label embeddings. In this paper we propose a deep version of the above-mentioned variant of neural network, where several hidden layers allow to take into account separately the different types of information given as input to the model. We evaluate our variant on the same tasks as the basic version. Results show that our variant is not only more effective than the other neural models, but also that it outperforms all the other models evaluated on the same tasks, reaching state-of-the-art performances.

MOTS-CLÉS : Réseaux neuronaux, apprentissage artificiel, compréhension de la parole, étiquetage de séquences.

KEYWORDS: Neural Networks, Machine Learning, Spoken Language Understanding, Sequence Labeling.

1 Introduction

Ces dernières années, les réseaux neuronaux (Jordan, 1989; Elman, 1990; Hochreiter & Schmidhuber, 1997) se sont montrés très efficaces dans toutes les tâches de TAL (Mikolov *et al.*, 2010, 2011; Collobert & Weston, 2008; Collobert *et al.*, 2011; Yao *et al.*, 2013; Mesnil *et al.*, 2013; Vukotic *et al.*, 2015). Les réseaux neuronaux récurrents notamment sont très efficaces grâce à leur architecture, qui permet de prendre en compte l'information passée comme contexte, et ce sous forme de représen-



(a) Jordan (b) I-RNN

FIGURE 1 – Le RNN de Jordan et le *I-RNN* utilisé dans cet article.

tations distributionnelles. Cependant la communauté s'est rendue compte très vite que ces modèles ne sont pas particulièrement adaptés à l'étiquetage de séquences textuelles. En effet, malgré leur relative efficacité, même les modèles les plus sophistiqués tels que LSTM et GRU (Hochreiter & Schmidhuber, 1997; Cho *et al.*, 2014) ne donnent pas des résultats significativement meilleurs que ceux obtenus avec des réseaux plus simples (Vukotic *et al.*, 2015, 2016; Dinarelli & Tellier, 2016a).

Ces modèles sont très efficaces pour *"garder en mémoire"* un contexte de mots arbitrairement long. Cependant cette capacité n'a pas été explicitement exploitée pour prendre en compte un contexte au niveau des étiquettes. Dans les tâches d'étiquetage de séquences, telles que l'annotation en partie du discours ou la compréhension automatique de la parole, le contexte des étiquettes prédites précédemment est au moins aussi important que le contexte au niveau des mots. C'est pour cette raison que les modèles CRF se sont montrés très efficaces sur ce type de tâches (Lafferty *et al.*, 2001; Hahn *et al.*, 2010; Vukotic *et al.*, 2015).

La fonction de décision classique des réseaux neuronaux (*softmax*) n'est pas adaptée à l'étiquetage de séquences. Pour éliminer cette limitation, plusieurs chercheurs ont proposé des modèles neuronaux hybrides *RNN+CRF* (Huang *et al.*, 2015; Lample *et al.*, 2016; Ma & Hovy, 2016), où la couche de sortie classique est remplacée par une couche CRF neuronale.

Dans les tâches d'étiquetage de séquences telles que celles mentionnées, des modèles très efficaces peuvent être conçus en modélisant les dépendances entre étiquettes. Nous reprenons pour cela l'idée de *I-RNN* décrite dans (Dinarelli & Tellier, 2016c). Ce modèle utilise des plongements pour les étiquettes, de la même façon que les plongements pour les mots, pour apprendre les dépendances entre étiquettes et leurs interactions. Les étiquettes prédites par le modèle sont converties en index qui sont donnés en entrée du réseau à l'étape de prédiction suivante, elles sont donc transformées en plongements de la même façon que les mots. Idéalement, cette architecture peut-être vue comme une évolution du réseau de Jordan (Jordan, 1989). Dans cette évolution, la connexion récurrente du réseau relie la couche de sortie à la couche d'entrée. Un schéma de ces deux architectures est montré dans la figure 1.

Le modèle *I-RNN*, bien qu'il utilise toujours une fonction de décision locale, s'est montré très efficace sur certaines tâches d'étiquetage de séquences. Le fait d'utiliser des plongements d'étiquettes permet à ce réseau de modéliser très efficacement les dépendances entre étiquettes, et d'obtenir des résultats meilleurs non seulement que les autres RNN, mais aussi que des modèles CRF très compétitifs.

Nous reprenons le modèle *I-RNN* (Dinarelli & Tellier, 2016c) comme modèle de base pour concevoir des modèles profonds encore plus efficaces. Nous exploitons dans cet article les avantages de l'apprentissage profond en utilisant deux couches cachées différentes : i) le premier niveau est divisé en plusieurs couches cachées pour apprendre séparément la représentation interne des différentes

entrées du réseau (mots, étiquettes et autres) ; ii) le second niveau prend en entrée la concaténation de la sortie des couches cachées précédentes et produit une nouvelle sortie utilisée par la couche de sortie pour prédire la prochaine étiquette.

Pour avoir une évaluation comparable, nous évaluons nos modèles sur les mêmes tâches que celles employées par les auteurs du modèle *I-RNN* : *ATIS* (Dahl *et al.*, 1994), en anglais, et *MEDIA* (Bonneau-Maynard *et al.*, 2006), en français. Grâce à la combinaison des plongements d'étiquettes pour apprendre les dépendances entre étiquettes, et aux couches cachées profondes pour apprendre des représentations internes sophistiquées, le modèle proposé dans cet article atteint des résultats à l'état-de-l'art sur les deux tâches.

Dans la suite de l'article, nous exposons d'abord (dans la section 2) la variante *I-RNN* utilisée comme modèle de base, puis la version profonde que nous proposons (section 3). Nous décrivons ensuite les expériences effectuées pour évaluer nos modèles (section 4), et nous concluons (section 5).

2 Le modèle *I-RNN*

Dans cet article, nous partons du modèle *I-RNN* proposé par (Dinarelli & Tellier, 2016c). Ce modèle utilise une table de correspondance pour les mots E_w et une pour les étiquettes E_l. Ces deux tables sont des matrices respectivement de taille $N \times D$ et $|O| \times D$, où N est la taille du dictionnaire de mots, D est la taille des plongements, $|O|$ est le nombre d'étiquettes qui correspond aussi à la taille de la couche de sortie du réseau.

Pour rendre le modèle plus efficace, une fenêtre de taille d_w est utilisée pour les mots, alors qu'elle est de taille d_l pour les étiquettes. Nous désignons par $E_w(w_i)$ le plongement d'un mot quelconque w_i, et par $E_l(y_i)$ le plongement d'une étiquette quelconque y_i. Les entrées du réseau au niveau des mots et des étiquettes à l'étape de traitement t, sont définies respectivement comme :

$$W_t = [E_w(w_{t-d_w})...E_w(w_t)...E_w(w_{t+d_w})]$$

$$L_t = [E_l(y_{t-d_l+1})E_l(y_{t-d_l+2})...E_l(y_{t-1})]$$

où [ ] désigne la concaténation de vecteurs (ou de matrices par la suite).

La couche cachée est ensuite calculée par :

$$h_t = \Phi(H[W_t L_t])$$

Φ étant la fonction d'activation *ReLU* dans cet article (Bengio, 2012). La sortie du réseau, c'est-à-dire l'étiquette prédite par le modèle, est enfin obtenue par :

$$y_t = softmax(Oh_t)$$

Grâce à l'utilisation des plongements d'étiquettes et à leur combinaison au niveau de la couche cachée, le modèle *I-RNN* peut apprendre de façon très efficace les dépendances entre étiquettes.

Comme on peut le remarquer, ce modèle n'est que idéalement récurrent, puisque les étiquettes prédites par le modèle sont converties en indexes, utilisés après pour sélectionner le plongement correspondant dans la matrice E_l. Aussi il n'y a pas de propagation en arrière à travers le temps (*BPTT*) puisque, comme dans (Dinarelli & Tellier, 2016a), le modèle est appris avec l'algorithme *Back-propagation* classique. Grâce à l'utilisation d'un contexte d'étiquette, tout cela donne une approximation d'un modèle récurrent appris avec l'algorithme *BPTT*. Comme il est montré dans (Mesnil *et al.*, 2013), ces choix ne semblent pas affecter significativement les résultats, du moins sur certains tâches.

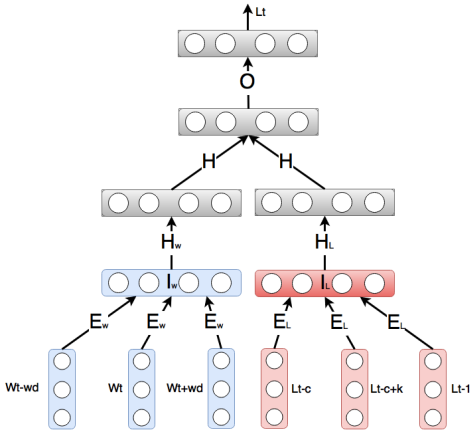


FIGURE 2 – Le *I-RNN* profond proposé dans cet article.

3 Réseaux profonds

Dans cet article nous proposons une variante profonde du modèle *I-RNN*. Cette variante exploite plusieurs niveaux de représentations internes au réseau, sous forme de couches cachées.

L'apprentissage profond pour le traitement d'images ou du signal audio (Hinton *et al.*, 2012; He *et al.*, 2015) a en effet montré qu'en utilisant plusieurs couches cachées, un réseau neuronal est capable d'apprendre des traits très fins et abstraits à partir des données. L'apprentissage profond a déjà, bien sûr, été utilisé avec succès dans les tâches de TAL. Mais quand l'entrée d'un tel réseau est constituée uniquement de mots, il est plus difficile de motiver l'utilisation des plusieurs couches cachées, car leur interprétation est problématique, au delà de l'évidence empirique constituée de meilleurs résultats. Puisque les réseaux utilisés dans cet article prennent en entrée au moins deux types différents d'informations (les mots et les étiquettes), l'utilisation de plusieurs couches cachées nous semble ici mieux justifiée.

Nous avons donc conçu un réseau neuronal profond avec deux niveaux de couches cachées. Au premier niveau chaque type d'entrée distinct (W_t et L_t, décrits plus haut) est connecté à sa propre couche cachée, respectivement à travers les paramètres H_w et H_l. La sortie de ces couches cachées est ensuite concaténée et donnée en entrée à une seconde couche cachée globale, elle-même utilisée comme entrée de la couche de sortie, comme déjà décrit plus haut. Un schéma de cette architecture profonde est fourni en figure 2.

Si d'autres entrées sont fournies au réseau (e.g. la sortie d'une convolution sur les caractères, comme cela sera proposé plus loin), celles-ci sont connectées à leur propre couche cachée dont la sortie est concaténée aux autres, avant d'être transmise en entrée à la seconde couche cachée.

La motivation de cette architecture profonde est de donner la possibilité au modèle d'apprendre une représentation interne différente pour chaque type d'entrée au premier niveau des couches cachées. Au second niveau, le réseau peut utiliser toute sa capacité de modélisation pour apprendre uniquement les interactions entre les entrées. Quand on utilise une seule couche cachée, le modèle est obligé d'apprendre à la fois une représentation interne globale pour toutes les entrées et pour leurs interactions, ce qui est un problème bien plus difficile.

MEDIA			ATIS		
Mots	**Classes**	**Étiquettes**	**Mots**	**Classes**	**Étiquettes**
Oui	-	Answer-B	i'd	-	O
l'	-	BDObject-B	like	-	O
hotel	-	BDObject-I	to	-	O
le	-	Object-B	fly	-	O
prix	-	Object-I	Delta	airline	airline-name
à	-	Comp.-payment-B	between	-	O
moins	relative	Comp.-payment-I	Boston	city	fromloc.city
cinquante	tens	Paym.-amount-B	and	-	O
cinq	units	Paym.-amount-I	Chicago	city	toloc.city
euros	currency	Paym.-currency-B			

TABLE 1 – Un exemple d'annotation pris du corpus MEDIA (gauche) et ATIS (droite).

4 Évaluation

4.1 Corpus pour la compréhension de la parole

Nous évaluons nos modèles sur les deux tâches de compréhension de la parole **ATIS** (*Air Travel Information System*) (Dahl *et al.*, 1994), en anglais, et **MEDIA** (Bonneau-Maynard *et al.*, 2006), en français.

Ces deux tâches sont celles employées pour l'évaluation des modèles auxquels nous nous comparons (Yao *et al.*, 2013; Mesnil *et al.*, 2013; Vukotic *et al.*, 2015, 2016; Dinarelli & Tellier, 2016a). Nous renvoyons les lecteurs à ces travaux pour plus de détails sur les corpus.

Un exemple comparatif d'annotation pris des deux corpus est montré dans le tableau 1.

4.2 Réglages

Notre variante de *I-RNN* profonde, appelée aussi *LD-RNN* dans (Dupont *et al.*, 2017), a été implémentée par nous-mêmes en *Octave* [1] avec le support de la bibliothèque *OpenBLAS*. [2]

Les modèles *I-RNN* sont entraînés avec la procédure suivante :

— deux modèles de langage neuronaux comme ceux de (Bengio *et al.*, 2003) sont entraînés pour générer les plongements des mots et des étiquettes. 30 et 20 époques d'entraînement, respectivement, sont utilisées pour ces deux modèles.

— les modèles *forward* et *backward* sont appris en utilisant les plongements entraînés à l'étape précédente. Ces deux modèles sont entraînés pendant 30 époques.

— le modèle bidirectionnel est appris en utilisant comme poids initiaux les modèles *forward* et *backward* entraînés à l'étape précédente. Ce dernier modèle est entraîné pendant 8 époques seulement. Grâce au fait qu'il est constitués de deux modèles appris à l'étape précédente, il est tout de suite très proche de l'optimum, souvent il est à l'optimum dès la première époque pour la tâche ATIS, alors qu'il atteint l'optimum entre la troisième et la cinquième époques sur la tâche MEDIA.

La première étape de la procédure est optionnelle, notamment sur des petites tâches comme ATIS

1. https ://www.gnu.org/software/octave/ ; Notre logiciel est décrit à la page http ://www.marcodinarelli.it/software.php et il est disponible sous requête.
2. http ://www.openblas.net

Modèle	Mesure F1		
	forward	backward	bidirectionnel
(Vukotic *et al.*, 2016) LSTM	95.12	–	95.23
(Vukotic *et al.*, 2016) GRU	95.43	–	95.53
(Dinarelli & Tellier, 2016b) E-RNN	94.73	93.61	94.71
(Dinarelli & Tellier, 2016b) J-RNN	94.94	94.80	94.89
(Dinarelli & Tellier, 2016b) I-RNN	95.21	94.64	94.75
I-RNN_{deep} Mots	94.47	94.29	94.52
I-RNN_{deep} Mots+Classes	95.40	**95.56**	**95.69**
I-RNN_{deep} Mots+Classes+CC	**95.72**	95.51	**95.67**

TABLE 2 – Comparaison des résultats sur la tâche ATIS avec la littérature, en terme de mesure F1.

Modèle	Mesure F1 / Concept Error Rate (CER)		
	forward	backward	bidirectionnel
(Vukotic *et al.*, 2015) CRF		86.00 / –	
(Hahn *et al.*, 2010) CRF		– / 10.6	
(Hahn *et al.*, 2010) ROVER×6		– / 10.2	
(Vukotic *et al.*, 2015) E-RNN	81.94 / –	– / –	– / –
(Vukotic *et al.*, 2015) J-RNN	83.25 / –	– / –	– / –
(Vukotic *et al.*, 2016) LSTM	81.54 / –	– / –	83.07 / –
(Vukotic *et al.*, 2016) GRU	83.18 / –	– / –	83.63 / –
(Dinarelli & Tellier, 2016b) E-RNN	82.64 / –	82.61 / –	83.13 / –
(Dinarelli & Tellier, 2016b) J-RNN	83.06 / –	83.74 / –	84.29 / –
(Dinarelli & Tellier, 2016b) I-RNN	84.91 / –	86.28 / –	86.71 / –
I-RNN_{deep} Mots	85.93 / 11.84	**86.80** / 10.50	**87.24 / 10.03**
I-RNN_{deep} Mots+Classes	85.63 / 11.87	86.64 / 10.41	**87.30 / 9.83**
I-RNN_{deep} Mots+Classes+CC	85.73 / 11.81	**86.83** / 10.33	**87.43 / 9.80**

TABLE 3 – Comparaison des résultats sur la tâche MEDIA avec la littérature, en terme de mesure F1 et *Concept Error Rate* (CER).

et MEDIA le pre-apprentissage des plongements n'apporte pas des gains significatifs. Cependant sur des tâches plus larges nous avons remarqué que même un modèle simple comme celui que nous utilisons pour pre-apprendre les plongements donne déjà des gains significatifs.

Puisque notre modèle, comme celui de (Dinarelli & Tellier, 2016a), n'est récurrent que grâce au fait qu'il utilise un contexte d'étiquettes sous forme de plongements, il est important d'utiliser au moins une étiquette prédite comme contexte. Nous avons optimisé la taille de ce contexte et nous avons trouvé que la meilleure taille est de 5 étiquettes passées pour les 2 tâches, bien que les différences par rapport à l'utilisation d'autres tailles ne soient pas très grandes. La taille du contexte des mots est de 7 pour MEDIA (3 à gauche, 3 à droite plus le mot courant) et de 11 pour ATIS. Ces valeurs correspondent à celles utilisées dans (Dinarelli & Tellier, 2016a), dont nous avons repris aussi les autres paramètres que nous n'allons donc pas discuter dans cet article.

4.3 Résultats

Tous les résultats de cette section sont des moyennes de 10 expériences. Les plongements des mots et des étiquettes ont été appris une seule fois pour toutes les expériences. Pour que nos résultats soit comparables à ceux obtenus avec d'autres modèles neuronaux, nous avons utilisé les mêmes réglages et les mêmes hyper-paramètres (Vukotic *et al.*, 2015, 2016; Dinarelli & Tellier, 2016a), et nous montrons aussi les résultats obtenus avec les modèles *forward*, *backward* et bidirectionnel.

Nos différents résultats ont été obtenus avec un nombre incrémental de types d'entrées : i) seulement les mots (*Mots*) ; ii) mots et classes de mots disponibles avec le corpus (*Mots+Classes*) ; iii) mots, classes et une convolution sur les caractères des mots, comme celle utilisée dans (Collobert *et al.*, 2011) (*Mots+Classes+CC*). Notre réseau figure dans les tableaux sous le nom I-RNN_{deep}. *E-RNN*,

J-RNN et *I-RNN* sont les réseau d'Elman, de Jordan et la version basique du *I-RNN* utilisée dans (Dinarelli & Tellier, 2016a) respectivement.

Les résultats obtenus sur la tâche ATIS sont montrés dans le tableau 2. Pour cette tâche, nous nous comparons avec les résultats de (Vukotic *et al.*, 2016; Dinarelli & Tellier, 2016a), qui ont aussi employé des couches cachées de type LSTM et GRU. Les résultats pour la tâche MEDIA sont montrés dans le tableau 3, dans laquelle nous nous comparons aux modèles décrits dans (Vukotic *et al.*, 2015, 2016; Dinarelli & Tellier, 2016a; Hahn *et al.*, 2010). Pour cette tâche nous montrons aussi l'évaluation en terme de *Concept Error Rate* (CER) [3], puisque certains travaux n'utilisent que cette mesure. Tous les autres résultats sont en terme de mesure *F1*.

A notre connaissance, le meilleur résultat jusqu'à présent pour la tâche ATIS était une mesure F1 de 95.53 obtenue par (Vukotic *et al.*, 2016) avec un modèle GRU. Les meilleurs résultats sur la tâche MEDIA étaient la mesure F1 de 86.71 atteinte par (Dinarelli & Tellier, 2016a) avec le modèle *I-RNN*, et la CER de 10.2 mentionnée par (Hahn *et al.*, 2010), avec une combinaison de 6 modèles individuels par *ROVER* ($ROVER \times 6$ dans le tableau) (Fiscus, 1997).

Tous nos résultats égaux ou meilleurs que l'état-de-l'art sont mis en gras dans les tableaux. Notre intuition qu'un réseau profond peut mieux apprendre les représentations internes des entrées et leurs interactions semble ainsi confirmée empiriquement. Sur ATIS, bien que nos modèles soient significativement meilleurs, tous les modèles sont relativement proches. Sur MEDIA, les écarts sont plus importants et nos modèles sont largement meilleurs que tous les autres. Le résultat le plus remarquable est la CER de 9.8 obtenue avec I-RNN_{deep} *Mots+Classes+CC*. Ce modèle améliore de 0.4 l'état-de-l'art obtenu par (Hahn *et al.*, 2010) avec le modèle $ROVER \times 6$, qui combinent 6 modèles individuels.

Au delà de ces résultats quantitatifs, une analyse simple des sorties de nos modèles montre qu'ils sont réellement capables d'apprendre efficacement les dépendances entre étiquettes : nos modèles ne font en effet aucune erreur de segmentation *BIO* (un O suivi d'un I, par exemple). Ceci suggère que l'utilisation des plongements d'étiquettes, combinée à l'utilisation de plusieurs couches cachées pour séparer l'apprentissage des représentations internes et de leurs interactions, pallie effectivement au caractère local de la fonction de décision.

5 Conclusion

Dans cet article nous avons décrit un modèle profond pour l'étiquetage de séquences dans le contexte de la compréhension automatique de la parole. Notre modèle s'appuie sur le modèle *I-RNN* introduit précédemment. Celui-ci utilise des plongements d'étiquettes pour apprendre leurs dépendances. Notre variante profonde, employant plusieurs couches cachées, permet d'apprendre séparément les représentations internes des entrées et leurs interactions. Les résultats obtenus sur les deux tâches ATIS et MEDIA sont à l'état-de-l'art, et confirment l'efficacité de cette solution.

Références

BENGIO Y. (2012). Practical recommendations for gradient-based training of deep architectures.

3. Il s'agit d'un taux d'erreur, le plus petit est le meilleur.

CoRR.

BENGIO Y., DUCHARME R., VINCENT P. & JAUVIN C. (2003). A neural probabilistic language model. *Journal of Machine Learning Research*, **3**, 1137–1155.

BONNEAU-MAYNARD H., AYACHE C., BECHET F., DENIS A., KUHN A., LEFÈVRE F., MOSTEFA D., QUGNARD M., ROSSET S. & SERVAN, S. VILANEAU J. (2006). Results of the french evalda-media evaluation campaign for literal understanding. In *LREC*, p. 2054–2059, Genoa, Italy.

CHO K., VAN MERRIENBOER B., GÜLÇEHRE Ç., BOUGARES F., SCHWENK H. & BENGIO Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*.

COLLOBERT R. & WESTON J. (2008). A unified architecture for natural language processing : Deep neural networks with multitask learning. In *Proceedings ICML*, p. 160–167 : ACM.

COLLOBERT R., WESTON J., BOTTOU L., KARLEN M., KAVUKCUOGLU K. & KUKSA P. (2011). Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, **12**, 2493–2537.

DAHL D. A., BATES M., BROWN M., FISHER W., HUNICKE-SMITH K., PALLETT D., PAO C., RUDNICKY A. & SHRIBERG E. (1994). Expanding the scope of the atis task : The atis-3 corpus. In *Proceedings of HLT Workshop* : ACL.

DINARELLI M. & TELLIER I. (2016a). Etude des reseaux de neurones recurrents pour etiquetage de sequences. In *Actes de la 23eme conference sur le Traitement Automatique des Langues Naturelles*, Paris, France : Association pour le Traitement Automatique des Langues.

DINARELLI M. & TELLIER I. (2016b). Improving recurrent neural networks for sequence labelling. *CoRR*.

DINARELLI M. & TELLIER I. (2016c). New recurrent neural network variants for sequence labeling. In *Proceedings of the 17th International Conference on Intelligent Text Processing and Computational Linguistics*, Konya, Turkey : Lecture Notes in Computer Science (Springer).

DUPONT Y., DINARELLI M. & TELLIER I. (2017). Label-dependencies aware recurrent neural networks. In *Proceedings of the 18th International Conference on Computational Linguistics and Intelligent Text Processing*, Budapest, Hungary : Lecture Notes in Computer Science (Springer).

ELMAN J. L. (1990). Finding structure in time. *COGNITIVE SCIENCE*, **14**(2), 179–211.

FISCUS J. G. (1997). A post-processing system to yield reduced word error rates : Recogniser output voting error reduction (ROVER). In *Proceedings of ASRU Workshop*, p. 347–352.

HAHN S., DINARELLI M., RAYMOND C., LEFÈVRE F., LEHEN P., DE MORI R., MOSCHITTI A., NEY H. & RICCARDI G. (2010). Comparing stochastic approaches to spoken language understanding in multiple languages. *IEEE TASLP*, **99**.

HE K., ZHANG X., REN S. & SUN J. (2015). Delving deep into rectifiers : Surpassing human-level performance on imagenet classification. In *IEEE ICCV*, p. 1026–1034.

HINTON G., DENG L., YU D., MOHAMED A., JAITLY N., SENIOR A., VANHOUCKE V., NGUYEN P., DAHL T. S. G. & KINGSBURY B. (2012). Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Processing Magazine*, **29**(6), 82–97.

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

HUANG Z., XU W. & YU K. (2015). Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv :1508.01991*.

JORDAN M. I. (1989). Serial order : A parallel, distributed processing approach. In *Advances in Connectionist Theory : Speech*. Erlbaum.

LAFFERTY J., MCCALLUM A. & PEREIRA F. (2001). Conditional random fields : Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML*, p. 282–289.

LAMPLE G., BALLESTEROS M., SUBRAMANIAN S., KAWAKAMI K. & DYER C. (2016). Neural architectures for named entity recognition. *arXiv preprint*.

MA X. & HOVY E. (2016). End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016*.

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

MIKOLOV T., KARAFIÁT M., BURGET L., CERNOCKÝ J. & KHUDANPUR S. (2010). Recurrent neural network based language model. In *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010*, p. 1045–1048.

MIKOLOV T., KOMBRINK S., BURGET L., CERNOCKY J. & KHUDANPUR S. (2011). Extensions of recurrent neural network language model. In *ICASSP*, p. 5528–5531 : IEEE.

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*, Dresde, Germany.

VUKOTIC V., RAYMOND C. & GRAVIER G. (2016). A step beyond local observations with a dialog aware bidirectional GRU network for Spoken Language Understanding. In *Interspeech*, San Francisco, United States.

YAO K., ZWEIG G., HWANG M.-Y., SHI Y. & YU D. (2013). : Interspeech.