Label-dependency coding in Simple Recurrent Networks for Spoken Language Understanding

Marco Dinarelli, Vedran Vukotic, Christian Raymond

Label-dependency coding in Simple Recurrent Networks for Spoken Language Understanding

Marco Dinarelli [1], *Vedran Vukotic*[2,3], *Christian Raymond*[2,3]

[1]Lattice, CNRS, ENS Paris, Université Sorbonne Nouvelle - Paris 3
PSL Research University, USPC (Université Sorbonne Paris Cité)
[2]INSA Rennes, France
[3]INRIA/IRISA, Rennes, France

marco.dinarelli@ens.fr,{vedran.vukotic,christian.raymond}@irisa.fr

Abstract

Modelling target label dependencies is important for sequence labelling tasks. This may become crucial in the case of Spoken Language Understanding (SLU) applications, especially for the slot-filling task where models have to deal often with a high number of target labels. Conditional Random Fields (CRF) were previously considered as the most efficient algorithm in these conditions. More recently, different architectures of Recurrent Neural Networks (RNNs) have been proposed for the SLU slot-filling task. Most of them, however, have been successfully evaluated on the simple ATIS database, on which it is difficult to draw significant conclusions. In this paper we propose new variants of RNNs able to learn efficiently and effectively label dependencies by integrating label embeddings. We show first that modeling label dependencies is useless on the (simple) ATIS database and unstructured models can produce state-of-the-art results on this benchmark. On ATIS our new variants achieve the same results as state-of-the-art models, while being much simpler. On the other hand, on the MEDIA benchmark, we show that the modification introduced in the proposed RNN outperforms traditional RNNs and CRF models.
Index Terms: recurrent neural networks, label dependencies,spoken language understanding, slot filling, ATIS, MEDIA

1. Introduction

In classical Spoken Language Understanding (SLU) systems, one of the key tasks is to label words with lexical semantics. For example, in the sentence "I want a Chinese restaurant near Tour-Eiffel", the word "Chinese" should be labeled as the food-type of a restaurant, and "Tour-Eiffel" as a relative place in Paris. Many algorithms have been investigated for slot tagging: SVM [1], HVS [2], Machine translation models, Finite State Transducers and Conditional Random Fields [3]. Recently, also Neural Networks have been investigated [4, 5, 6]. Neural networks have the advantage to come together with new text representations. Discrete items in the text are mapped into vectors, named often embeddings, using popular word embedding methods [7, 8]. This representation has several advantages, the most salient one is to make words that are syntactically or semantically related, close to each-other in the representation space. This ability is particularly useful for several tasks, but not particularly for SLU on the ATIS task because the database of this task already provides important clusters (*e.g.* city names, airline names, places, *etc.*). Anyway, this representation appears to be noise robust [6]. Neural networks are not dedicated sequence-labelling algorithms and many efforts have been made to improve their ability to process sequences. Recurrent Neural Network (RNN) architectures like LSTM [9] have been investigated to better model long range dependencies in the observations. RNNs like Jordan architectures have been proposed to better model target label sequences [5]. In this work we focus on modeling the target label dependencies. We propose a modification of the Jordan architecture by introducing an embedding of the previous predicted target labels. This simple modification results in a RNN very effective at learning label dependencies, and allows improvements over the other RNNs proposed in the literature as well as state-of-the-art CRF models.

Unfortunately, the public widely used benchmark ATIS [10, 11] is not very challenging and a wide variety of methods provides similar (very good) results, including methods that are not specifically designed for sequence labeling. These last methods fail [12, 3, 6] when evaluated on MEDIA [13], another public SLU database where modeling label-dependencies is crucial to obtain good results. This indicates that conclusions formulated from results obtained on the ATIS database are not particularly strong. We will provide in this work results from experiments conducted on both databases.

Results on the MEDIA task, in particular, provide evidence to conclude that: i) the proposed variant of RNN using label embeddings outperforms by a large margin the standard Jordan RNN, and thus also the Elman RNN which is less effective than the Jordan model [6]; ii) by simply using label embeddings, RNNs can model label dependencies more effectively compared to RNNs using a CRF neural layer like the one used in [5, 14, 15]; iii) the proposed variant of RNN provides the new state-of-the-art results on both the ATIS and the MEDIA tasks.

We particularly stress on results obtained on the MEDIA task because, as it has been shown and as we will show in this paper, only models keeping label-dependencies into account obtain good results on this task, which means in turn that these models are the most suited for sequence labeling.

2. Datasets

In our experiments we used two datasets: ATIS and MEDIA. ATIS is a publicly available corpus used in the early nineties for SLU evaluation. MEDIA has been collected in the last decade and is available through ELRA since 2008.

2.1. ATIS

The Air Travel Information System (ATIS) task [10] is dedicated to provide flight information. The semantic representation used is frame based. The SLU goal is to find the good frame and fill the corresponding slots.

The training set consists of 4978 utterances selected from the Class A (context independent) training data in the ATIS-2 and ATIS-3 corpora while the ATIS test set contains both the ATIS-3 NOV93 and DEC94 datasets. Please see [10] for more details.

2.2. MEDIA

The research project MEDIA [13] evaluates different SLU models of spoken dialogue systems dedicated to provide tourist information. A corpus made of 1250 French dialogues has been collected by ELDA, following a Wizard of Oz protocol: 250 speakers have followed 5 hotel reservation scenarios. This corpus has been transcribed manually and annotated with concepts from a rich semantic ontology. The representation is based on the definition of concepts that can be associated with 3 kinds of information. First a concept is defined by a label and a value; for example with the concept date, the value 2006/04/02 can be associated. Second, a specifier can be attached to a concept in order to link the concept, and to go from flat concept/value representations to hierarchical ones; for example, the concept date can be associated with the specifiers *reservation* and *begin* to specify that this date is the beginning date of a hotel reservation. Third, modal information is added to each concept (positive, affirmative, interrogative or optional). Table 1 shows an example of dialogue turn from the MEDIA corpus with only concept-value information. The first column contains the segment identifier in the message, the second column shows the chunks W^c supporting the concept c of the third column. In the fourth column the value of the concept c in the chunk W^c is displayed. The MEDIA semantic dictionary contains 83 concept labels, 19 specifiers and 4 types of modal information. In this study we will focus only on concept extraction. No specifiers, values or modal information are considered, so the tag-set consists of 83 labels. The MEDIA corpus is split into 3 parts. The first part (720 dialogues, 12K messages) is used for training the models, the second (79 dialogues, 1.3K message) is used for selecting the best system, and the third part (200 dialogues, 3.4K message) is used as test.

3. Simple Recurrent Networks

3.1. Elman network

Elman networks have been proposed in [16] and are defined as:

$$\begin{aligned} h_t &= \sigma_h(W_h x_t + U_h h_{t-1} + b_h) \\ y_t &= \sigma_y(W_y h_t + b_y) \end{aligned}$$

n	W^c	c	**value**
1	yes	answer	yes
2	the	RefLink	singular
3	hotel	BDObject	hotel
4	which	null	
5	price	object	payment-amount
6	is below	comparative-payment	below
7	fifty five	payment-amount-int	55
8	euros	payment-currency	euro

Table 1: *Example of message with concept+value information. The original French transcription is: "oui l'hôtel dont le prix est inférieur à cinquante cinq euros"*

where x_t is the input vector, h_t the hidden layer vector, y_t the output vector, W, U and b are parameter matrices and vector, σ_h and σ_y are activation functions.

Elman RNNs use the previous hidden state as contextual information (h_{t-1}), but they don't use any information about previous predicted labels. For this reason, while they have shown good results as any other model on the ATIS task, they are among the least effective neural models on the MEDIA task [6].

3.2. Jordan network

Jordan networks have been proposed in [17] and are defined as follows:

$$\begin{aligned} h_t &= \sigma_h(W_h x_t + U_h y_{t-1} + b_h) \\ y_t &= \sigma_y(W_y h_t + b_y) \end{aligned}$$

This model uses previous predicted labels as contextual information to predict the current label. However previous labels are provided as input to the hidden layer either as raw network outputs, or as *one-hot* representations[1]. Raw network outputs are the output of the softmax output layer [18], which computes a probability distribution over all possible labels defined in the task. One-hot representations can be computed from raw outputs putting 1 at the position corresponding to the maximum probability, and zero anywhere else.

3.3. eJordan

The improved RNN proposed in this paper is based on a similar idea as the one described in [19, 20].

In this variant predicted labels are mapped into embeddings, the same way as words. Word embeddings are stored in a matrix $E_w \in \mathbb{R}^{|D_w| \times N}$, where $|D_w|$ is the size of the word dictionary, N is the size chosen for the embeddings. In the same way label embeddings are stored in a matrix $E_l \in \mathbb{R}^{|D_l| \times N}$, where $|D_l|$ is the size of the label dictionary, which is also the size of the network output y_t.

In order to keep notation lighter, we indicate with y_t both the raw output of the network (computed by the softmax) and the one-hot representation of the label. The latter can be seen in turn as the index of the corresponding label. With this formalism, the input of the hidden layer is $x_t = E_w(w_t)$, like in the other RNNs and w_t is the word to be labeled at position t in a sequence, and $z_t = E_l(y_{t-1})$, which is the embedding of the previous predicted label y_{t-1}. The hidden and output layers are then computed as:

$$\begin{aligned} h_t &= \sigma_h(W_h x_t + U_h z_{t-1} + b_h) \\ y_t &= \sigma_y(W_y h_t + b_y) \end{aligned}$$

As we can see thus, the only difference between the proposed variant and a Jordan RNN is that in our variant the label used as contextual information is provided as an embedding. For this reason we name our variant *eJordan*, for *embedded Jordan* RNN.

3.4. Bi-eJordan

Since a couple of years, RNNs are provided as bidirectional models [21, 5]. These models allow to keep into account both past and future information to predict the current label.

[1] *one-hot* representations are sparse vectors representing dictionary entries. The entry having index i in a dictionary V of size $|V|$, is represented with a vector of size $|V|$ which is zero everywhere, except at position i where it has value 1.

We provide also our eJordan variant as bidirectional model. As described in [21], in this variant we use first a backward model to predict labels in backward direction, that is from the end to the begin of a sentence. Such labels are then used as future predicted labels by a bidirectional model, which processes sentences in forward direction and computes its final output as the geometric mean of the forward and backward decisions:

$$y_t = \sqrt{y_t^f \odot y_t^b}$$

where y_t^f is the output of the forward model, y_t^b is the output of the backward model, and $\odot$ is the element-wise product.

4. Experimental protocol

We will compare our eJordan model against several competitor architectures:

- the basic Multi-Layer Perceptron (MLP) with softmax output layer (no recurrence), also named Feed-Forward Neural Network (FFNN) in the literature, in order to show the importance of modeling target label dependencies. This is called MLP+SOFT in later.
- the Jordan RNN: in order to compare the difference between one-hot and fine tuned embedded representation
- a MLP with CRF layer on top instead of the softmax (and obviously applied on sequences), called later MLP+CRF

A boosting based [22] system is also presented. This system is a local classification model not designed at all for sequence labeling like MLP+SOFT and uses only symbolic features (no embedding). This model and the MLP+SOFT are used to illustrate the difference in results that can be obtained when modeling label dependencies and sequences is important, like in MEDIA, with respect to the ATIS task, where any of the described models reaches state-of-the-art results.

4.1. Features and configuration

One of the objective of this paper is to fairly compare systems and their ability to model target label dependencies. So, usual and reasonable configurations previously published are used and fixed for all neural systems. Thus they are evaluated and compared to each other in the same conditions, which are:

- observation: word or class if the word belongs to a semantic class (*e.g.* CITY_NAME)
- size of observation window: 7 for MEDIA and 11 for ATIS
- hidden layer: 200 for MEDIA and 100 for ATIS
- size of the embedding: 200 for MEDIA, 100 for ATIS

All systems have been ran 10 times for 30 epochs. Averages of 10 results will be provided in terms of:

- F1 measure computed by the script conlleval[2]: this measure tends to show how good is the segmentation of concepts over surface forms.
- Concept Error Rate (CER) measure computed by sclite[3] on the target label level: this measure tends to show how good is the concept recognition in the perspective of using SLU in spoken dialog systems.

[2] `http://www.cnts.ua.ac.be/conll2000/chunking/output.html`

[3] `http://www1.icsi.berkeley.edu/Speech/docs/sctk-1.2/sclite.htm`

Algo	Parameters	conlleval F1	CER sclite
ATIS			
boosting	i=2500 d=2	95.69	5.0
MLP+SOFT	151,488	95.67 (0.07)	5.00 (0.09)
MLP+CRF	157,606	95.45 (0.11)	5.28 (0.12)
Bi-Jordan	338,376	95.69 (0.07)	4.97 (0.07)
Bi-eJordan	340,576	95.74 (0.02)	4.91 (0.03)
MEDIA			
boosting	i=3500 d=3	77.11	18.2
MLP+SOFT	642,135	83.60 (0.16)	12.71 (0.21)
MLP+CRF	660 360	86.34 (0.19)	10.96 (0.14)
Bi-Jordan	1,399,882	86.15 (0.09)	11.12 (0.11)
Bi-eJordan	1,743,082	86.97 (0.12)	10.34 (0.19)
BiGru+CRF	2,328,360	86.69 (0.13)	10.13 (0.21)

Table 2: *Performances of the various algorithms on ATIS and MEDIA. Averaged (over 10 runs) F1 measure (%), Concept Error Rate (%) and their respective standard deviations (in parenthesis).*

Boosting systems boost bonsai trees [22], number of iterations (i) and depth (d) of the trees are fixed arbitrarily to some values that provide good results on the used benchmarks (they are used to set our expected low baseline since they don't use neither embeddings nor sequence dedicated mechanisms). This system is doing automatically feature selection, thus a larger context (observation window of 20) is provided and allows improvements.

The MLP+SOFT, Jordan and eJordan models have been implemented in *Octave*[4]. The other neural models have been implemented in *Keras*[5].

5. Results

The first remarkable result in table 2 is the performance of the boosting system: on ATIS, this system is performing very well, as well as all other algorithms despite the fact that it is not using any embedding and has no knowledge about target label dependencies. On the opposite, on MEDIA, this system looks largely ineffective in comparison to the others. These results illustrate clearly the fact that ATIS is not a challenging task. It illustrates also the fact that it is not possible to draw strong conclusions about the fact that an algorithm is better or not than another one: almost every algorithm is able to provide outstanding results on ATIS, and noise may be a better explanation for the slight difference between algorithms than the effectiveness of the algorithm itself [23, 6, 24].

As shown in the example in table 1, MEDIA is a much more challenging task: first, the semantic annotation is richer; second, labels are segmented over multiple words, which can create relatively long label dependencies and increases in practice the number of labels to be recognized to 135 (using the *BIO* segmentation formalism); third, though it is not specifically addressed in this paper, coreference phenomena are annotated in the MEDIA task, making annotation decisions depend on long past contexts. An idea of the difficulty of this task with respect to ATIS is given also by the absolute magnitude of results in table 2 (9-12 F1 points lower than results on ATIS).

Comparing results in table 2 on MEDIA among the neural models, provides evidence of interesting outcomes.

[4] https://www.gnu.org/software/octave/; The code is described at http://www.marcodinarelli.it/software.php and available upon request

[5] https://keras.io

First we note that models integrating increasingly rich information on label dependencies provide increasingly good results: the MLP+SOFT model, which has no label information, is the less effective; the traditional Jordan RNN integrates the previous label as one-hot representation, outperforming by a large margin the MLP model; the MLP+CRF further improves results, showing that it can integrate longer range label information thanks to the global-level probability normalization of CRF; very interestingly, the most effective model on MEDIA is the proposed eJordan as bidirectional variant (Bi-eJordan), which uses label embeddings. Since this variant uses a local decision function (the softmax), from these results we can deduct that the use of label embeddings, together with their combination with word embeddings at the hidden layer, allows RNNs to model more fine label dependency features and word-label interactions than a CRF neural layer, overcoming in fact the limitation of using a local decision function.

Second, eJordan achieves a CER of 10.34 on average, and 10.32 according to more accurate model on the development data on the 10 runs. These results can be compared to [3] on only *attributes* extraction. To the best of our knowledge this is the best result achieved on this task with an individual model[6].

Finally, we give more insights on the behavior of neural models when integrating in different ways and at different degrees contextual label information. For this purpose we simply analyze results in terms of accuracy on void concepts (O) compared to accuracy on all the other concepts ($\neg O$). Indeed the ratio of void concepts is very different on the two tasks addressed in this paper: $35,623$ out of $52,170$ (68,28%) on ATIS, and $33,186$ out of $95,851$ (34,62%) on MEDIA. Also, while in the ATIS corpus there is no segmentation of concepts over multiple words (each concept is instantiated by one token), in the MEDIA task, on the opposite, concepts are segmented over relatively long lexical chunks.

As consequence we expect models not aware of label dependencies to be somehow naive, predicting correctly a larger amount of void concepts (to minimize the risk). This is the consequence of the large representation of this category in the training data, combined to the fact that these models cannot "trade" the decision conducted from word-level information with the one conducted from label-level information. In contrast, the more sophisticated the representation of label context information in the neural model, the more we expect the model to be effective in predicting labels other than the void concept. In these models the bias toward predicting the over-represented class of void concepts can be possibly in contrast with the constraints introduced by label dependencies.

This simple analysis is depicted in table 3. We can see once again that all models perform astonishingly well on ATIS, and even more astonishingly close: all models achieve accuracy close or higher than 99 on void concepts, and higher than 97 on the other concepts.

The same analysis on MEDIA is much more interesting. As expected, the MLP+SOFT model, which is the only one without any contextual label information, achieves a relatively high accuracy on void concepts (the second best), while it performs the worst on the other concepts, and by more than 1 point from the second best (Jordan). We can consider the other 3 neural models addressed in this analysis, as more and more sophisticated in integrating contextual label information, the order being Bi-Jordan, Bi-eJordan and MLP+CRF. The accuracy on non-void concepts reflects indeed this ranking. The Bi-eJordan variant however reaches a better compromise between accuracy on void and non-void concepts, and it is thus the most effective among these 4 neural models in terms of F1 measure and CER[7] (table 2).

Model	Accuracy on O	Accuracy on ¬O
ATIS		
MLP+SOFT	99.01	97.16
MLP+CRF	98.99	97.16
Bi-Jordan	99.01	97.21
Bi-eJordan	99.01	97.27
MEDIA		
MLP+SOFT	95.89	87.98
MLP+CRF	93.73	89.04
Bi-Jordan	96.46	88.01
Bi-eJordan	94.68	88.60

Table 3: *Comparative results of the different neural models described in the paper in terms of accuracy on void concepts (*O*) and all the other concepts (*¬O*). Models with less label-level contextual information are those with higher accuracy, but lower F1 and CER.*

We would like to point out that eJordan and the CRF mechanism must not be considered in mutual exclusion. In [14, 15] we can see actually that the CRF neural layer used so far is somehow complementary to eJordan, in the sense that it does not represent labels as embeddings. The combination of these two models may thus lead to even more sophisticated models.

However the goal of this work is not to produce the best result on the addressed benchmarks, but to propose and compare some label-dependencies aware methods for SLU in a fair way. Of course, better architectures may be easily proposed: for example, using LSTM as hidden layer to better encode long context input may allow further improvements. We started investigating also these more complex models, in particular a bidirectional GRU [25] with a CRF neural layer as output layer. Preliminary results are given in table 2 with the name *BiGRU+CRF*. Also richer inputs may be provided to the networks, *e.g.* word embeddings externally trained on huge amount of data, character-level convolution like in [14, 15], and so on.

6. Conclusion

We proposed in this paper a recurrent neural network architecture to better model target label dependencies in sequence labeling problems for SLU. This architecture, named eJordan, is a slight modification of the Jordan network where the label predicted at time $t-1$ is embedded and injected as input to the network at time t. A bidirectional eJordan network is fairly compared and outperforms traditional competitors, Jordan and MLP+CRF on the MEDIA task. As usual, every methods tends to perform similarly (and very well) on the ATIS dataset.

7. Acknowledgment

This work has been partially funded by the French ANR project Democrat ANR-15-CE38-0008[8]. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the GTX Titan X GPU used for this research.

[6]The best absolute result in [3] is 10.2, but it is obtained with a combination of 6 individual models

[7]We recall the reader that F1 measure and CER don't take void concepts into account.

[8]http://www.agence-nationale-recherche.fr/?Projet=ANR-15-CE38-0008

8. References

[1] T. Kudo and Y. Matsumoto, "Chunking with Support Vector Machines," in *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics on Language Technologies*, ser. NAACL '01. Stroudsburg, PA, USA: Association for Computational Linguistics, 2001, pp. 1–8. [Online]. Available: http://dx.doi.org/10.3115/1073336.1073361

[2] Y. He and S. Young, "Semantic Processing using the Hidden Vector State Model," *Computer Speech and Language*, vol. 19, pp. 85–106, 2005.

[3] S. Hahn, M. Dinarelli, C. Raymond, F. Lefèvre, P. Lehnen, R. De Mori, A. Moschitti, H. Ney, and G. Riccardi, "Comparing Stochastic Approaches to Spoken Language Understanding in Multiple Languages," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 6, pp. 1569–1583, August 2011.

[4] G. Mesnil, X. He, L. Deng, and Y. Bengio, "Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding," in *INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, Lyon, France, August 25-29, 2013*, 2013, pp. 3771–3775. [Online]. Available: http://www.isca-speech.org/archive/interspeech_2013/i13_3771.html

[5] G. Mesnil, Y. Dauphin, K. Yao, Y. Bengio, L. Deng, D. Hakkani-Tur, X. He, L. Heck, G. Tur, D. Yu *et al.*, "Using recurrent neural networks for slot filling in spoken language understanding," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 23, no. 3, pp. 530–539, 2015.

[6] V. Vukotic, C. Raymond, and G. Gravier, "Is it time to switch to word embedding and recurrent neural networks for spoken language understanding?" in *InterSpeech*, Dresde, Germany, September 2015.

[7] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," in *International Conference on Learning Representations*, 2013.

[8] R. Lebret and R. Collobert, "Word Embeddings through Hellinger PCA."

[9] K. Yao, B. Peng, Y. Zhang, D. Yu, G. Zweig, and Y. Shi, "Spoken language understanding using long short-term memory neural networks," in *Spoken Language Technology Workshop (SLT), 2014 IEEE*. IEEE, 2014, pp. 189–194.

[10] D. A. Dahl, M. Bates, M. Brown, W. Fisher, K. Hunicke-Smith, D. Pallett, C. Pao, A. Rudnicky, and E. Shriberg, "Expanding the scope of the ATIS task: the ATIS-3 corpus," in *HLT*, 1994, pp. 43–48.

[11] C. Raymond and G. Riccardi, "Generative and Discriminative Algorithms for Spoken Language Understanding," in *InterSpeech*, Antwerp, Belgium, August 2007, pp. 1605–1608.

[12] S. Hahn, P. Lehnen, C. Raymond, and H. Ney, "A comparison of various methods for concept tagging for spoken language understanding," in *Proceedings of the Language Resources and Evaluation Conference*, Marrakech, Morocco, May 2008.

[13] H. Bonneau-Maynard, S. Rosset, C. Ayache, A. Kuhn, and D. Mostefa, "Semantic Annotation of the French Media Dialog Corpus," in *InterSpeech*, Lisbon, September 2005.

[14] X. Ma and E. Hovy, "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*, 2016.

[15] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," *arXiv preprint arXiv:1603.01360*, 2016.

[16] J. L. Elman, "Finding structure in time," *COGNITIVE SCIENCE*, vol. 14, no. 2, pp. 179–211, 1990.

[17] M. I. Jordan, "Serial order: A parallel, distributed processing approach," in *Advances in Connectionist Theory: Speech*, J. L. Elman and D. E. Rumelhart, Eds. Hillsdale, NJ: Erlbaum, 1989.

[18] Y. Bengio, "Practical recommendations for gradient-based training of deep architectures," *CoRR*, vol. abs/1206.5533, 2012. [Online]. Available: http://arxiv.org/abs/1206.5533

[19] M. Dinarelli and I. Tellier, "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), Avril 2016.

[20] D. Bonadiman, A. Severyn, and A. Moschitti, "Recurrent context window networks for italian named entity recognizer," *Italian Journal of Computational Linguistics*, vol. 2, 2016. [Online]. Available: http://disi.unitn.it/moschitti/since2013/2016_IJCoL_Moschitti_NER-CNNs-IT.pdf

[21] M. Schuster and K. Paliwal, "Bidirectional recurrent neural networks," *Trans. Sig. Proc.*, vol. 45, no. 11, pp. 2673–2681, nov 1997. [Online]. Available: http://dx.doi.org/10.1109/78.650093

[22] A. Laurent, N. Camelin, and C. Raymond, "Boosting bonsai trees for efficient features combination : application to speaker role identification," in *InterSpeech*, Singapour, September 2014. [Online]. Available: http://bonzaiboost.gforge.inria.fr

[23] K. Yao, G. Zweig, M.-Y. Hwang, Y. Shi, and D. Yu, "Recurrent Neural Networks for Language Understanding," in *InterSpeech*. Interspeech, August 2013. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=200236

[24] G. Tur, D. Hakkani-Tur, and L. Heck, "What is left to be understood in ATIS?" in *Spoken Language Technology Workshop (SLT), 2010 IEEE*. IEEE, 2010, pp. 19–24.

[25] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, *Empirical evaluation of gated recurrent neural networks on sequence modeling*, 2014.