Joint syntactic and semantic analysis with a multitask Deep Learning Framework for Spoken Language Understanding

Jeremie Tafforeau, Frédéric Béchet, Thierry Artières, Benoit Favre

Joint syntactic and semantic analysis with a multitask Deep Learning Framework for Spoken Language Understanding

Jeremie Tafforeau[1], *Frederic Bechet*[1]
Thierry Artiere[1,2], *Benoit Favre*[1]

[1]Aix-Marseille Université, CNRS, LIF UMR 7279, 13000, Marseille, France
[2]Ecole Centrale de Marseille, 13000, Marseille, France
firstname.lastname@lif.univ-mrs.fr

Abstract

Spoken Language Understanding (SLU) models have to deal with Automatic Speech Recognition outputs which are prone to contain errors. Most of SLU models overcome this issue by directly predicting semantic labels from words without any deep linguistic analysis. This is acceptable when enough training data is available to train SLU models in a supervised way. However for open-domain SLU, such annotated corpus is not easily available or very expensive to obtain, and generic syntactic and semantic models, such as dependency parsing, Semantic Role Labeling (SRL) or FrameNet parsing are good candidates if they can be applied to noisy ASR transcriptions with enough robustness. To tackle this issue we present in this paper an RNN-based architecture for performing joint syntactic and semantic parsing tasks on noisy ASR outputs. Experiments carried on a corpus of French spoken conversations collected in a telephone call-centre are reported and show that our strategy brings an improvement over the standard pipeline approach while allowing a lot more flexibility in the model design and optimization.

Index Terms: Spoken Language Understanding , Recurrent Neural Networks , Long Short Term Memory , FrameNet parsing , Multitask

1. Introduction

Natural Language Understanding (NLU) is the process of producing semantic interpretations from words and other linguistic events that are automatically detected in a text conversation or a speech signal. For Spoken Language Understanding, hierarchical shallow semantic models are widely used, consisting on determining first the domain, then the intent, and finally the slot-filling entities needed to fulfill a query [1]. Domain, intent and slot labels are directly linked to the application targeted: personal assistant, web queries, etc. The drawback of using application-specific labels is the need of an annotated corpus of sufficient size in order to perform supervised learning.

For Open Domain NLU, generic purpose semantic models can be used, such as FrameNet or Abstract Meaning Representation (AMR). Once this generic meaning representation is obtained, a translation process can be applied for projecting generic predicates and concepts to application-specific ones. This kind of approach can help reducing the need of annotated corpus for training NLU models, however their main drawback is the need for fine grain parsing processes involving syntactic dependency parsing or semantic role labeling.

For such tasks, the standard linguistic processing pipeline is made of a chain of sequential processes such as POS tagging, chunking, Named-Entity (NE) recognition, syntactic parsing and semantic analysis. This architecture is clearly sub-optimal when processing Automatic Speech Recognition (ASR) output which are prone to contain errors: each error at a given level can lead to more errors at the next level, following a *snow ball* effect. This phenomenon is very critical when processing spontaneous speech in spoken conversations because of the high word error rate of ASR systems on such data.

Recently, approaches based on a continuous vector space representation for words and deep neural networks have been proposed to unify several NLP tasks into a single model that can be optimized selectively according to the application targeted [2]. In the SLU domain, DNN architectures have been used for domain and intent classification [1] and slot-filling [3].

If some consistent gains over sequence tagging methods such as Conditional Random Fields (CRF) have been reported on the SLU benchmark corpus ATIS [3], these gains are rather limited compared to the big boost of performance observed in the image and acoustic classification communities when embeddings and DNN were introduced. However three main characteristics make DNN-based models good candidates for building NLU models:

1. the use of large amount of unlabeled data for learning word representation when dealing with limited amount of in-domain data [4];
2. the joint optimization of DNN over several NLP tasks;
3. the ability of Recurrent Neural Networks (RNN) to maintain contextual information through sequence decoding with a memory model such as the *Long Short Term Memory* model [5].

We propose in this study an SLU model following these principles dedicated to jointly perform syntactic and semantic analysis through a multi-task [6] approach. This model is a bidirectional RNN with LSTM. It takes as input word embeddings that can be learned on a large out-of-domain corpus. We show on a corpus of speech conversations over phone the advantages of such an approach over a standard sequential pipeline for high Word Error Rate (WER) transcriptions.

2. Related Work

Bidirectional Recurrent Neural Network have the ability to model context in two directions, from left-to-right and right-to-left, in order to capture long dependencies that can occur either before or after a current target. These models, introduced by [7] have been used for speech recognition [8] or sequence tagging [9].

Multitask learning focuses on learning different yet related tasks simultaneously [6] with a common classifier. The global cost function for multitask learning is a sum of costs of each of the individual tasks. This strategy has been applied to Spoken Language Understanding in order to learn simultanously several semantic annotation schemes that share slots [10]. In the context of Deep Neural Network, multitask has been proposed by [11] in order to have a unified representation and architecture for several NLP tasks such as POS tagging, chunking, Named Entity tagging, Semantic Role Labelling. Recently the same paradigm has been applied to Machine Translation [12] and Automatic Speech Recognition [13].

Unlike [10], the SLU multitask model we propose here is not dedicated to output different semantic annotations sharing slots but rather different NLP tasks, all related, belonging to several linguistic levels such as POS tagging, syntactic dependency parsing, Named Entity tagging and FrameNet parsing. This is rather similar to [11] and [2] where a unified representation is used over task. One difference of our model is the explicit weighting of each task in minimizing the loss at each iteration and the use of bi-directional RNN models with LSTM.

The SLU task consists in detecting semantic frames and frame elements in the automatic transcriptions of spoken conversations with a relatively high Word Error Rate (WER). We introduced this task in [14] where we used a syntactic dependency parser adapted to process spontanous speech [15]. This approach worked well on manual transcriptions or when the WER is low. In this paper we remove the need for such a parser with our RNN multitask architecture in order to increase the robustness of our system for high WER transcriptions.

3. Tasks & Corpus

We use in this study the *RATP-DECODA*[1] corpus. It consists of 1514 conversations over the phone recorded at the Paris public transport call-centre over a period of two days [16]. The calls last 3 minutes on average, representing a corpus of about 74 hours of signal.

Several levels of linguistic annotations have been performed on the manual transcriptions of this corpus: Part-Of-Speech, Named Entities, disfluencies, syntactic dependencies. These annotations have been performed with a semi-supervised process described in [17, 15] and based on the MACAON NLP pipeline [18]. In addition to these annotations a semantic frame annotation scheme, based on FrameNet, has been applied on the syntactic parses of the corpus [19]. We did not perform a full-text annotation of the corpus, but rather selected a set of *Lexical Units* (LU) relevant to the corpus domain and perform annotation of the frames and frame elements triggered by these LUs in the corpus.

In our experiments, the semantic frame annotations are projected at the word level: each word is either labeled as `null` if it is not part of a frame realization, or as the name of the frame (or frame elements) it represents. In our training corpus, 28% of the words have a non-null semantic label and there are 336 different frame labels. A lot of ambiguities come from the disfluencies which are occurring in this very spontaneous speech corpus.

An example of annotated corpus is given in table 1 for the sentence: *I I lost my phone in bus 38*. The semantic frame labels given at the word level follow a simple syntax: **position** (either B for *begin* or I for *inside*); **frame name**; **role** (either *agent*, *object* or LU for *lexical unit*). There are 335 different labels to predict at the word level, corresponding to 208 frame segment labels, representing 71 different frames. As expected the top frames are related either to the transport domain (SPACE) or the communication domain (COM and COG).

[1]The RATP-DECODA corpus is available at the Ortolang SLDR data repository: `http://sldr.org/sldr000847/fr`

Table 1: *Example of syntactic and semantic annotation at the word level in the RATP-DECODA corpus*

id	*word*	*POS*	*disf*	*NE*	*dep*	*link*	*frame*
1	I	prp	rep	–	disf	2	–
2	I	prp	–	–	sbj	3	B_losing_agent
3	lost	vbp	–	–	root	0	B_losing_LU
4	my	prp$	–	–	nmod	5	B_losing_obj
5	phone	nn	–	–	obj	3	I_losing_obj
6	in	in	–	–	loc	5	B_losing_obj
7	bus	nn	–	B_tran	pmod	6	I_losing_obj
8	38	cd	–	I_tran	mod	7	I_losing_obj

In our setting the semantic frame annotations are strongly linked to the syntactic ones since it is the dependency parses which have been used to link frame triggers and frame elements. The main advantage of this approach to semantic annotation is to lower the need for manual supervision if reliable syntactic annotations are available. The projection from syntactic parse to semantic frames and frame elements is a rule-based system described in [19]. However this method leverage the need for syntactic parsing, which can be difficult when processing noisy ASR transcriptions.

The ASR trasncriptions used in this study are described in [20]. They are obtained thanks to the LIUM system based on the Kaldi decoder [21] with DNN acoustic models as well as LIUM rescoring tools [22]. The average WER is 34.5%. This high error rate is mainly due to speech disfluencies and noisy acoustic environments.

Because of this high WER, we believe that our multitask approach is a good candidate to perform a joint syntactic and semantic analysis. This model is presented in the following section.

4. Neural network framework

4.1. Bidirectional LSTM Network

In sequence tagging task, we have access to both past and future input features for a given time, we can thus use a bidirectional LSTM network (Figure 1) as proposed in [8]. In doing so, we can efficiently make use of past features (via forward states) and future features (via backward states) for a specific time frame.

The basic idea of bidirectional recurrent neural nets is to present each training sequence forwards and backwards to two separate recurrent nets, both of which are connected to the same output layer. This means that for every point in a given sequence, the Bi-LSTM has complete, sequential information about all points before and after it. Also, because the net is free to use as much or as little of this context as necessary, there is no need to find a (task-dependent) time-window or target delay size. The recurrent layer is composed of LSTM cells in order to solve the gradient vanishing/exploding problems [23, 24]. Its input is the word sequence of interest and associated morphological features. Word inputs are encoded via a lookup table which associate words to low-dimensional embedding vectors learned over the training. As morphological features, we consider boolean values about capitalisation, numbers and non alpha-numeric characters presence. In addition, we also use a

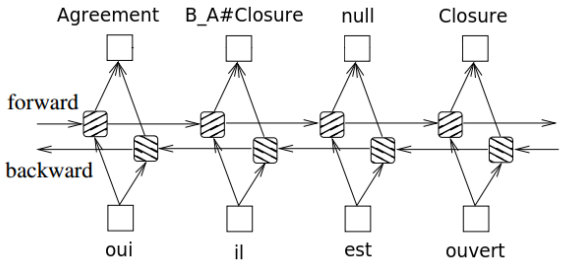

Figure 1: *A End-to-End Bi-LSTM network in a frame detection context. The same word sequence is fed to two distinct RNN, both connected to the same fully-connected output layer, in order to predict the best frame labels sequence.*

word representation based on a bag of character bi-grams. For example, the word "boat" is represented as $\langle$ *bo, ba, bt, oa, ot, at* $\rangle$. This model is learned with Back-Propagation Thought Time [25] and Adam optimizer [26] using a log-likelihood criterion with a decaying learning rate. We use dropout regularization with a firing rate $p = 0.5$.

5. Multitask Learning

5.1. Neural Architecture

In order to allow a multitask learning, we added task-specific fully-connected output layers as presented in Figure 2. This layers are all fed with the same Bi-LSTM state and are trained to independently predict their associated task label. By doing so, every parameters of the network (except the task-specific output layers) are shared over tasks. In our experiments, we consider Part-of-Speech tagging, Disfluencies detection, Named Entities recognition, and Syntactic parsing as additional tasks of interest.

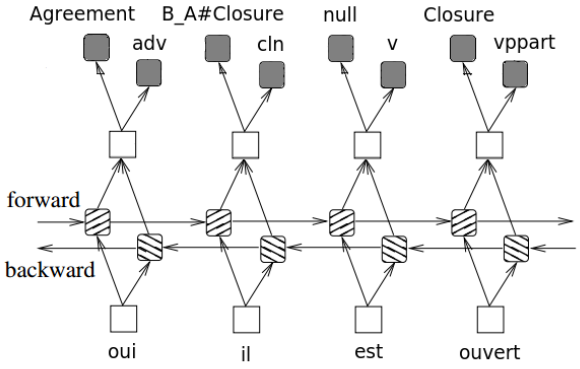

Figure 2: *A Multitask Bi-LSTM network. The same Bi-LSTM state is feed to task-specific output layers. In order to preserve readability, we only represent Part-of-Speech tagging and Frame detection output layers.*

5.2. Learning Criterion

Multitask learning focuses on learning several tasks simultaneously with a single classifier. The global cost function is a weighted sum of individual task costs. More formally, considering E_k the log-likehood criterion of the k-th task, we compute a multitask learning criterion E combining task-specific losses.

$$E = \sum_{k=1}^{N} \alpha_k E_k = \sum_{k=1}^{N} \alpha_k \sum_{x \in X} \log p(y_k | x, \theta_k, \theta_s)$$

with $\sum \alpha_k = 1$, and where θ_s are shared parameters between tasks and θ_k are the k-th task-specific parameters. This allows to jointly train the network on N several tasks of interest while optimizing a unified objective function. In our experiments, we first study a uniform α combination weights ($\forall k\ \alpha_k = \frac{1}{N}$). In this configuration, each task is learned with the same level of interest. On an other hand, we study the influence of our main task weight. While increasing it, the network focus more and more on the semantic frame detection task at the expense of additional syntactic tasks. As exposed in Table 3, best frame detection performances were obtained with a main task weight eight times higher than additional tasks ones.

6. Experiments

The experiments reported in this paper have been done on the RATP-DECODA corpus annotated with semantic frames as presented in section 3. This corpus has been split into 3 partitions: train, dev and test described in table 2.

Table 2: *Description of the train, dev and test partition of the RATP-DECODA corpus used for models training and evaluation.*

part.	#dialog	#turn	#word	%word in frames
train	1243	76158	495451	28.8%
dev	144	9074	60968	28.7%
test	100	3347	23258	29.2%

Only the test partition has been fully manually checked for annotation errors. The train and dev partitions has been annotated thanks to the semi-supervised method presented in [17, 15]. All the hyper-parameters of our model have been tuned on the dev corpus.

The main task evaluated here is the detection of frame segments. For example, in the annotated sentence of Table 1, there are 4 segments to detect: **losing_agent** (*I*), **losing_LU** (*lost*), **losing_obj** (*my phone*), **losing_obj** (*in bus 38*). For each frame segment found by an automatic system we check if a segment with the same label occur at the same time stamp $\pm\delta$ (with $\delta = 2s$). Similarly we look for every reference segment in the automatic output. Precision, Recall and F-measure are computed for this frame segment detection tasks and are reported in this section.

Our experimental results are presented in Table 3 and 4. Five systems are compared: two *end-to-end* systems predicting directly semantic frames from word sequences; two pipeline systems (PL) implementing a sequence of NLP tasks (POS tagging, disfluency detection, NE recognition, dependency parsing) before predicting frames; and one fully integrated multitask system (MT) predicting all NLP level with the semantic frames directly from the word sequence.

- **E2E:CRF**: a Conditional Random Field model trained in an end-to-end fashion to predict semantic frames only from word features;
- **ST:BiRNN**: trained in an end-to-end fashion. Inputs are words projected as word embeddings and learning targets are frame labels. In our experimental setting, we

Table 3: *Experimental results table on manual transcriptions of the corpus test set for a Single Task (ST) setting (ST:BiRNN) then for our multitask approach (MT:BiRNN) with two distribution weights over the tasks (uniform and biased toward frame detection). We report Part-of-Speech and Syntactic Dependencies accuracy (in %). For each other task of interest, namely Disfluencies, Named Entity and Frame detection, we measure our system performances with Precision, Recall and F1-measure followed by the standard deviation.*

Model	POS	Syntax	Disfluency	NER	Frame
ST:BiRNN	95.7(0.02)	78.8(0.11)	95.1 80.6 87.2(0.12)	96.5 91.4 93.9(0.17)	79.6 86.7 82.9(0.03)
MT:BiRNN 1/1/1/1/1	95.6(0.01)	79.0(0.15)	96.5 81.8 88.5(0.03)	95.7 93.6 94.6(0.12)	80.7 85.7 83.2(0.03)
MT:BiRNN 1/1/1/1/8	95.5(0.02)	78.4(0.01)	97.4 81.7 88.8(0.29)	94.7 94.1 94.4(0.23)	80.6 86.2 83.4(0.06)

use 256-dimension word embeddings and both recurrent layers are composed of 512 LSTM cells.

- **PL:MACAON**: a state-of-the-art dependency parser based on the MACAON pipeline [18] adapted to process spontaneous speech transcriptions [15]. The semantic frames are obtained thanks to the same rule-based system as the one used to label the manual transcriptions of the corpus and described in [19].
- **PL:CRF**: a Conditional Random Field model taking as input features produced by the MACAON pipeline (POS, disfluency, NE and dependency labels).
- **MT:BiRNN**: our RNN model trained to predict semantic frames as well as POS tag, disfluencies, named entity and syntactic annotations.

The results presented in table 4 show that our BiRNN systems outperform the pipeline systems on both reference and ASR transcriptions. It is interesting to notice that the **ST:BiRNN** is almost as good as the **MT:BiRNN**, although no syntactic information was given to the network during training. The pipeline system **PL:CRF** is not as good as **PL:MACAON**. This can be explained by the fact that the reference semantic annotations were done using a syntactic parser (manually corrected on the test corpus). Therefore this semantic task is closely link to a syntactic parsing task, and it is not surprising that a state-of-the-art parser performs better to find distant dependencies than a CRF with local features.

The robustness of our systems toward ASR errors is explored in Figure 3. We split the test corpus dialogs in partitions corresponding to the WER obtained by their automatic transcriptions. We defined 5 partitions at 20, 30, 40, 50 and over 50% WER. The results of the 3 systems **PL:MACAON**, **PL:CRF** and **MT:BiRNN** are compared. As we can see, for all level of WER, the multitask system **MT:BiRNN** is more robust than the two pipeline systems.

Table 4: *Experimental results on frame and frame element detection on manual transcription and ASR output of the test corpus. The standard deviation for the F-measure results of our BiRNN models is given in brackets.*

trans.	ref. transcriptions			ASR (WER=34.5)		
metric	P	R	F	P	R	F
E2E:CRF	78.4	72.5	75.3	74.2	44.0	55.3
ST:BiRNN	80.4	85.6	82.9 (0.03)	70.9	53.9	61.3 (0.06)
PL:MACAON	79.6	84.2	81.8	69.4	52.4	59.7
PL:CRF	78.4	80.4	79.4	72.5	48.9	58.4
MT:BiRNN	80.8	86.0	**83.4** (0.06)	71.0	54.2	**61.5** (0.10)

Finally we analyze in table 3 the impact of our multitask setting **MT:BiRNN** compared to a single-task approach **ST:BiRNN** by modifying the weights distribution over tasks of our learning criterion as described in section 4. As we can see, with an equal weighting, a small improvement can be observed with the MT setting. By increasing the weight of the semantic frame task, a further improvement is achieved for the main task with a minimal impact on other additional tasks.

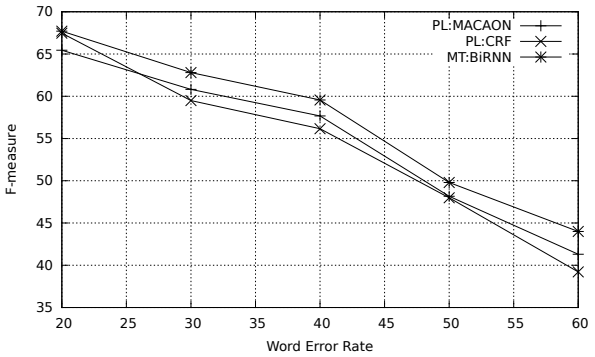

Figure 3: *F-measure of frame and frame element detection according to the WER of dialog ASR transcriptions for three systems: two pipelines and our bidirectional RNN multitask*

7. Conclusions

We present in this paper an RNN-based architecture for performing joint syntactic and semantic parsing tasks on noisy ASR outputs. Experiments carried on a corpus of French spoken conversations over phone collected in a call-centre are reported and show that our strategy brings an improvement over the standard pipeline approach while allowing a lot more flexibility in the model design and optimization. Although multitask processing does not bring a significant improvement, it is worth noticing that the main advantage of this approach is to produce a rich annotation on several linguistic levels at no extra-cost, unlike traditional pipeline approaches.

8. Acknowledgements

- The research leading to these results has received funding from the European Union (FP7) under grant agreement 610916 – SENSEI.
- The Tesla K40 used for this research was donated by the NVIDIA Corporation.
- The authors would like to thank the LIUM team for sharing ASR transcriptions of the RATP-DECODA corpus.

9. References

[1] G. Tur, L. Deng, D. Hakkani-Tur, and X. He, "Towards deeper understanding deep convex networks for semantic utterance classification." IEEE International Confrence on Acoustics, Speech, and Signal Processing (ICASSP), March 2012. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=164624

[2] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," in *the Journal of Machine Learning Research 12*, 2011, pp. 2461–2505.

[3] 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, pp. 3771–3775.

[4] 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*, 2015.

[5] M. Sundermeyer, H. Ney, and R. Schlüter, "From feedforward to recurrent LSTM neural networks for language modeling," *IEEE/ACM Transactions on Audio, Speech & Language Processing*, vol. 23, no. 3, pp. 517–529, 2015. [Online]. Available: http://dx.doi.org/10.1109/TASLP.2015.2400218

[6] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, Jul. 1997. [Online]. Available: http://dx.doi.org/10.1023/A:1007379606734

[7] 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

[8] A. Graves, A. Mohamed, and G. E. Hinton, "Speech recognition with deep recurrent neural networks," *CoRR*, vol. abs/1303.5778, 2013. [Online]. Available: http://arxiv.org/abs/1303.5778

[9] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging," *CoRR*, vol. abs/1508.01991, 2015. [Online]. Available: http://arxiv.org/abs/1508.01991

[10] X. Li, Y.-Y. Wang, and G. Tur, "Multi-task learning for spoken language understanding with shared slots." Annual Conference of the International Speech Communication Association (Interspeech), August 2011. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=172324

[11] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th International Conference on Machine Learning*, ser. ICML '08. New York, NY, USA: ACM, 2008, pp. 160–167. [Online]. Available: http://doi.acm.org/10.1145/1390156.1390177

[12] D. Dong, H. Wu, W. He, D. Yu, and H. Wang, "Multi-task learning for multiple language translation," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, 2015, pp. 1723–1732. [Online]. Available: http://aclweb.org/anthology/P/P15/P15-1166.pdf

[13] A. Mohan and R. C. Rose, "Multi-lingual speech recognition with low-rank multi-task deep neural networks," in *ICASSP*, 2015.

[14] F. Bechet, A. Nasr, and B. Favre, "Adapting dependency parsing to spontaneous speech for open domain spoken language understanding," in *Interspeech, Singapore*, 2014.

[15] A. Nasr, F. Bechet, B. Favre, T. Bazillon, J. Deulofeu, and A. Valli, "Automatically enriching spoken corpora with syntactic information for linguistic studies," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014), Reykjavik, Iceland, May 26-31, 2014.*, 2014, pp. 854–858.

[16] F. Bechet, B. Maza, N. Bigouroux, T. Bazillon, M. El-Beze, R. D. Mori, and E. Arbillot, "Decoda: a call-centre human-human spoken conversation corpus," in *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*. European Language Resources Association (ELRA), may 2012.

[17] T. Bazillon, M. Deplano, F. Bechet, A. Nasr, and B. Favre, "Syntactic annotation of spontaneous speech: application to call-center conversation data." in *LREC*, 2012, pp. 1338–1342.

[18] A. Nasr, F. Béchet, J. Rey, B. Favre, and J. Le Roux, "Macaon: An nlp tool suite for processing word lattices," *Proceedings of the ACL 2011 System Demonstration*, pp. 86–91, 2011.

[19] J. Trione, F. Bechet, B. Favre, and A. Nasr, "Rapid FrameNet annotation of spoken conversation transcripts," in *Proceedings of the 11th Joint ACL-ISO Workshop on Interoperable Semantic Annotation*, 2015.

[20] C. Lailler, A. Landeau, F. Bchet, Y. Estve, and P. Delglise, "Enhancing the ratp-decoda corpus with linguistic annotations for performing a large range of nlp tasks," in *Proceedings of LREC*, 2016.

[21] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. EPFL-CONF-192584. IEEE Signal Processing Society, 2011.

[22] P. Deléglise, Y. Estève, S. Meignier, and T. Merlin, "The lium speech transcription system: a cmu sphinx iii-based system for french broadcast news." in *Interspeech*, 2005, pp. 1653–1656.

[23] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994. [Online]. Available: http://www.iro.umontreal.ca/ lisa/pointeurs/ieeetrnn94.pdf

[24] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, vol. 6, no. 2, pp. 107–116, Apr. 1998. [Online]. Available: http://dx.doi.org/10.1142/S0218488598000094

[25] M. Boden, "A guide to recurrent neural networks and backpropagation," in *IN THE DALLAS PROJECT, SICS TECHNICAL REPORT T2002:03, SICS*, 2002.

[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980