Can we predict how challenging Spoken Language Understanding corpora are across sources, languages and domains?

Frederic Bechet, Christian Raymond, Achraf Hamane, Rim Abrougui, Gabriel Marzinotto, Géraldine Damnati

Can we predict how challenging Spoken Language Understanding corpora are across sources, languages and domains?

Frederic Bechet[1], Christian Raymond[2], Achraf Hamane[1], Rim Abrougui[3], Gabriel Marzinotto[3], Géraldine Damnati[3]

Abstract State-of-the art Spoken Language Understanding models of Spoken Dialog Systems achieve remarkable results on benchmark corpora thanks to the winning combination of pretraining on large collection of out-of-domain data with contextual Transformer representations and fine tuning on in-domain data. On average, performances are almost perfect on benchmark datasets such ATIS. However some phenomena can affect greatly these performance, like unseen events or ambiguities. They are the major sources of errors in real-life deployed systems although they are not necessarily equally represented in benchmark corpora. This paper aims to predict and characterize error-prone utterances and to explain what makes a given corpus more or less challenging. After training such a predictor on benchmark corpora from various languages and domains, we confront it to a new corpus collected from a French deployed vocal assistant with different distributional properties. We show that the predictor can highlight challenging utterances and explain the main complexity factors even though this corpus was collected in a completely different setting.

1 Introduction

In the Transformer era, Spoken Language Understanding models of Spoken Dialog Systems have achieved remarkable results on a wide range of benchmark tasks. State of the art models involve contextual embeddings trained on a very large quantity of out-of-domain text, usually with a Transformer approach, followed by a fine-tune training process on in-domain data to generate the semantic representation required, often made of intent+concept/value labels [10].

[1]Aix Marseille University, CNRS, LIS UMR 7020 - e-mail: frederic.bechet@lis-lab.f, achraf.hamane@etu.univ-amu.fr
[2] INSA Rennes - IRISA - Rennes - e-mail: christian.raymond@irisa.fr
[3] Orange Labs, Lannion - e-mail: firstname.lastname@orange.com

This winning strategy gives a boost in performance compared to previous models, mostly because of the generalization power of pretrained contextual embeddings. However, if on some SLU benchmark corpora like ATIS, such models have reached almost perfect performance, other corpora remain challenging and performance can be greatly affected by the amount and the quality of data available for training or by the complexity and ambiguity of the semantic annotation scheme.

But how can we characterize how challenging a corpus is? What are the factors that explain why some utterances still resist to Transformer-based models? And can we predict automatically this complexity when dealing with a new corpora in order to partition data into several sets representing different sources and levels of difficulty?

Moreover, it was noticed in [3] and [9] that standard benchmark datasets don't contain enough *difficult* examples that can be found in real-life deployed services, giving a false impression that there are no margin of improvement in current models. Furthermore, the distribution of utterances in benchmark corpora doesn't necessarily reflect real-life usage. Distributions in corpora collected from deployed services are more likely to be imbalanced, with on one hand possibly more *easy* utterances that researchers may not consider interesting to integrate in benchmark corpora and on the other hand a larger variety of complex phenomena that are under-represented in benchmark corpora.

This paper aims to give some answers to these questions on benchmark SLU corpora as well as a new dataset collected from a deployed voice assistant in order to verify if knowledge extracted on *artificial* data can generalize to *real* human-machine interactions.

2 Predicting corpus complexity

To predict corpus complexity we follow the approach proposed in [1] and [2] inspired by the NIST *Recognizer output voting error reduction* [7] method for scoring Automatic Speech Recognition (ASR) performance. In this method, multiple recognizers output are combined by voting on each decision, the most probable one being the output with most votes. This method acknowledges the fact that there is some kind of uncertainty in the output produced by statistically trained models, therefore using multiple decisions can help increasing robustness in the decision process. This phenomenon is particularly true for current deep learning models which involve some randomness in parameter initialization, leading to produced different performance on different runs of the same model.

In [1] it was proposed to use a modified version of the ROVER method in order to qualify each utterance of an evaluation corpus for an SLU task of semantic concept recognition seen as a sequence labeling problem. By running multiple SLU models on the same data, we obtain several concept recognition hypotheses at the word level. According to the agreement between hypotheses, a cluster label is given to each word : **AC** means that all models agree, and the output is correct; **AE** means

i	*word* w_i	*label(ref,u,i)*	*label*(m_1,*u*,*i*)	*label*(m_2,*u*,*i*)	*cluster*
1	find	O	O	O	**AC**
2	flights	O	O	O	**AC**
3	from	O	O	O	**AC**
4	new-york	B-from-city	B-from-city	**B-from-city**	**AC**
5	new-york	O	**B-from-city**	**B-to-city**	**NE** $\rightarrow$ **NCE**
6	next	B-date-dep	**B-date-dep**	**O**	**NC** $\rightarrow$ **NCE**
7	saturday	I-date-dep	**I-date-dep**	**B-date-arr**	**NC** $\rightarrow$ **NCE**

Table 1 Example of annotation of utterance u with two SLU models (m_1, m_2) and the resulting cluster for each word

that all models agree, and the output is incorrect; **NC** means that some models disagree but at least one of them is correct; **NE** means that some models disagree but none of them is correct. It was hypothesized in [2] that cluster **AC** corresponds to the easy samples, **NC** to the difficult ones, **NE** to the very difficult ones, and finally **AE** to the *problematic* ones, often corresponding to annotation errors.

In this study we want to go further than just qualifying a sample as *easy* or *difficult* by understanding the reason behind this qualification. Moreover we want to uncover generic principles, that can be applied to any SLU task, independently from the language, the topic or the semantic model related to a given corpus. To this purpose we propose the following method based on a 2-step process:

First step:

1. Select a set of L SLU corpora, with concept annotation at the word level (with B,I,O info if multi-word concepts), partitioned into train, development and test.
2. Select a set of N Deep Neural Network (DNN) sequence tagger implementing different DNN architectures and using different kinds of word pretraining.
3. Train the N sequence taggers separately on each train partition of the L corpora, and evaluate the performance on their corresponding development and test sets.
4. Label each word in the development and test corpora with the **AC**, **AE**, **NC** and **NE** labels according to the agreement and the correctness of the concept label predicted by the N concept sequence taggers;

An example of such process is given in table 1 for two SLU concept taggers. Since this utterance contains at least one word labeled **NCE**, it will belong to the **NCE** cluster containing the *difficult* utterances.

The second step of the process aims at understanding what makes a sample *easy* or *difficult*. **AC** samples stand for the *easy* one while labels **AE**, **NC** and **NE** are grouped into a new label **NCE** for *difficult* samples.

Second step:

1. Describe each word in the development and test corpora of each SLU corpus with language independent, topic independent and concept independent features (*Generic Features - GF*), such as syntactic features and features related to the coverage of the training corpus (e.g. *how many times this word has been seen with this label in the training corpus?*).

2. Train a *glass-box* classifier such as Adaboost on the union of the *L* development corpora described by *GF* features to predict the complexity labels **AC** and **NCE** and evaluate its performance on the SLU test corpora also labeled with **AC** and **NCE** labels as in step 1.4.
3. Analyze the classification model obtained by uncovering the rules and their weights automatically learned on *GF* features to predict label **NCE** in order to qualify the major complexity factors on all the SLU corpora considered.

At the end of this 2-step process we obtain a *complexity* classifier that can process any new SLU corpus, regardless of its language, topic and semantic model, as long as each word is described by *GF* features, without the need to train and evaluate any SLU system. This classifier labels each word with a complexity label (**AC** or **NCE**), a score, and an *explanation* about this complexity, obtained by analyzing the **NCE** rules learned and their weights. This kind of *explanation* is obtained by characterizing each feature type in the *GF* set. This is presented in the next section.

3 Analyzing complexity factors

To analyze utterance complexity with respect to an SLU task such as concept tagging, we make the following assumption, following previous work done on Named Entity Recognition [3, 9]: the two main sources of complexity that can affect an SLU model are *ambiguity* and lack of *coverage* of the training corpus.

- **ambiguity**: an utterance can be ambiguous if a word or a sequence of words can correspond to multiple labels in the semantic model and if either there is not enough context to help removing the ambiguity, or if the underlying structure of the utterance is complex (long utterance, multiple verbs, disfluencies, ...);
- **coverage**: this source of complexity comes from a lack of coverage between the training and the evaluation data. The most obvious phenomenon is *Out-Of-Vocabulary* words, but it can also comes from a new or a rare association between a known word and a label, or a new n-gram of known words.

The features we use in the *GF* set to describe a word *W* with label *l* in a sentence *S* are either related to ambiguity or coverage. They are defined in table 2.

All the syntactic features are obtained through a parsing process on the train, dev and test partitions of each corpus. In order to be language independent, we use parsers [12] based on the *Universal Dependency* syntactic model [13]. Hence syntactic features are shared across languages. Once a corpus is projected into the *GF* feature set, there is no lexical information and no semantic labels left, therefore corpora on different languages, topics and semantic models can be merged in order to train the *complexity* classifier for producing the **AC** or **NCE** labels.

We use a glass-box classifier called *Bonzaiboost*[1] [8] based on boosting [14] where a set of weak classifiers made of small decision trees on the features of *GF*

[1] `http://bonzaiboost.gforge.inria.fr/`

Ambiguity
of semantic labels acceptable for W
of Part-Of-Speech (POS) acceptable for W + POS label
of possible syntactic dependency for W + dependency label
distance between W and the sentence syntactic root.
utterance length (in words)
% of words in S belonging to a concept
Coverage
of occurrences of W in train
of occurrences of (W,l) in train
is bigrams $(W-1,W)$ and $(W,W+1)$ occurring in train?

Table 2 The Generic Feature (GF) set

corpora	**ATIS**	**MEDIA**	**SNIPS**	**Djingo_Spk**
#word	8333	25977	6595	34938
#sent	893	3005	700	9984
vocabulary	485	1219	1752	2637
#concept	84	70	39	34
#intent	-	-	7	109
%OOD sentences	0	0	0	6.6%
%sent $\in$ *train* $\cap$ *test*	1.9	44.6%	0.9%	76.9%
%sent+concept	99.3%	86.5%	100%	59.3%
av. sent length	10.3	7.6	9.16	4.2

Table 3 Corpora characteristics

are weighted in order to predict the output labels. When processing a sentence, the set of rules matching the input features are selected and the label chosen is the one maximizing the score according to the rules weights. When the **NCE** label is predicted, we can check in the selected rules which ones have contributed positively to predict the *difficult* label. Since each rule belongs either to the *ambiguity* or *coverage* set, we can estimate the % of weight in the **NCE** score that belongs to either set, and thus *explain* if this difficulty comes from an ambiguity issue or lack of coverage in the training data.

The classifier outputs decision at the word level, however they can be projected at the sentence level with this simple rule: the *easy* utterances are those where all words have been labeled as **AC**; the *difficult* utterances are those containing at least one word labeled as *NCE*. Therefore we can use the *complexity* classifier output in order to select utterances with a certain level of difficulty, expressed by the **NCE** score, and belonging either to the *ambiguity* or *coverage* category.

4 Experiments on benchmark corpora

The method presented in the two previous sections has been implemented on 4 SLU benchmark corpora described in table 3 split into train, dev and test partitions:

pretraining	bigru	gru	self attention
BERT	M1	M3	M5
random	M2	M4	M6

Table 4 Description of models M1 to M6 in terms of pretraining conditions and DNN architecture

Model/F-measure	ATIS	MEDIA	SNIPS	M2M
M1	94.6	85.7	95.4	91.5
M2	93.8	81.7	69.6	91.7
M3	94.7	85.8	95.2	93.6
M4	79.0	60.1	69.0	91.0
M5	94.8	85.3	95.9	93.0
M6	77.4	59.8	68.9	91.0

Table 5 Concept detection performance (F-measure) for models M1...M6 on the 4 benchmark corpora

1. M2M: this corpus is a fusion of two datasets containing dialogues for restaurant and movie ticket booking. It has been released by [15] and collected using their M2M framework (Machines Talking To Machines) that combines dialogue self-play and crowd sourcing to generate dialogues.
2. ATIS: The Air Travel Information System (ATIS) task [6] is dedicated to provide flight information.
3. MEDIA: this corpus is made of 1250 French dialogue, dedicated to provide tourist information. It 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 [4].
4. SNIPS: this corpus has been collected by the SNIPS company. It is dedicated to 7 in-house tasks, SearchCreativeWork, GetWeather, BookRestaurant,PlayMusic, AddToPlaylist,RateBook, SearchScreeningEvent [5].

In order to obtain the complexity labels **AC** and **NCE**, we developed 6 SLU sequence tagger models (M1...M6) in order to predict concept labels at the word level on our 4 corpora. These 6 systems differ either by the pretraining condition (BERT or random initialization) and the DNN architecture (GRU, BIGRU or self-attention) as described in table 4 . These systems follow state-of-the-art architectures for SLU concept tagging [10]. If BERT pretraining outperforms by a large margin random initialization, it is interesting to keep this option for detecting easy utterance that does not need any generalization capabilities outside the training data. Table 5 shows F-measure results obtained by all systems on the four corpora.

As we can see models without pretraining (M2, M4 and M6) obtain much worst performance on all corpora except M2M, first indication that this corpus does not need generalization capabilities.

From the automatic labelling with models M1 to M6, we can compute labels AC and NCE at the word and sentence levels as presented in section 2. The repartition between *easy* (AC) and *difficult* (NCE) utterances is presented in table 6 . We can

label/%	ATIS	MEDIA	SNIPS	M2M
AC (word)	89.8%	70.1%	83.1%	96.1%
NCE (word)	10.2%	29.9%	16.9%	3.9%
AC (sent)	46.2%	54.3%	35.1%	84.2%
NCE (sent)	53.8%	45.7%	64.9%	15.8%

Table 6 Repartition into *easy* (AC) and *difficult* (NCE) samples at the word and sentence levels

label/Fmes	ATIS	MEDIA	SNIPS	M2M
AC	98.7	98.5	99.7	99.0
NCE)	91.7	82.3	93.1	68.6

Table 7 Performance of model M1 on AC and NCE sentences

see that the amount of *difficult* tokens and sentences differ greatly from one corpus to another, giving more insights about the complexity of a given corpus than just looking at the average SLU performance. For example, although the M2M corpus seems more challenging that ATIS and SNIPS according to the best model (M1) in table 5, we can see in table 6 that it contains a lot more of *easy* tokens and sentences than the other corpora.

Table 7 clearly indicates the relevance of the AC/NCE clustering since performance obtained with a state-of-the-art model such as M1 obtain much worse results on NCE utterances compared to AC utterances.

Following the method presented in section 3, we trained a *Bonzaiboost* classifier to predict the complexity labels **AC** and **NCE** on the union of the 4 development corpora. The results are presented in table 8. As we can see, if the classification results vary according to the corpus considered, we obtain an F-mesure over 93% for label *AC* and almost 60% on label NCE. These are encouraging results considering that no lexical nor semantic labels are used as features to predict utterance complexity and that we mix in the training and test conditions very different SLU corpora on different languages, topics and semantic models.

Table 9 shows the analysis of the NCE decisions in terms of the respective weights of the *ambiguity* and *coverage* features as described in section 3. As we can see it is interesting to notice that, depending on the corpus considered, the complexity can come mostly because of coverage issues (ATIS and M2M), ambiguity issues (MEDIA) or a mixt of both (SNIPS). The distribution obtained on partitions obtained with predicted labels, rather than reference ones are very similar. This is also encouraging showing that even if the complexity classifier makes errors (60% Fmeasure), it can still be used to accurately partition a corpus according to criteria linked to the utterance complexity and the sources of this complexity.

ATIS	**Precision**	**Recall**	**F-measure**
AC	91.75	98.26	94.89
NCE	60.61	23.26	33.61
MEDIA	**Precision**	**Recall**	**F-measure**
AC	82.55	87.82	85.11
NCE	63.03	52.80	57.46
SNIPS	**Precision**	**Recall**	**F-measure**
AC	92.54	96.04	94.26
NCE	58.93	42.31	49.25
M2M	**Precision**	**Recall**	**F-measure**
AC	98.08	99.89	98.98
NCE	97.00	65.10	77.91
All corpora			
all	**Precision**	**Recall**	**F-measure**
AC	**91.58**	**95.57**	**93.53**
NCE	**68.42**	**52.21**	**59.23**
All	**88.83**	**88.83**	**88.83**

Table 8 Classification performance on AC/NCE labels with the GF feature set. Training on the union of all corpora.

ATIS	**weight(NCE,AMBIG)**	**weight(NCE,COVER)**
reference	13.1%	86.9%
prediction	19.9%	80.1%
MEDIA	**weight(NCE,AMBIG)**	**weight(NCE,COVER)**
reference	84.4%	15.6%
prediction	84.3%	15.7%
SNIPS	**weight(NCE,AMBIG)**	**weight(NCE,COVER)**
reference	37.2%	62.8%
prediction	23.5%	76.5%
M2M	**weight(NCE,AMBIG)**	**weight(NCE,COVER)**
reference	4.1%	95.9%
prediction	2.3%	97.7%
all	**weight(NCE,AMBIG)**	**weight(NCE,COVER)**
reference	**65.8%**	**34.2%**
prediction	**68.0%**	**32.0%**

Table 9 % of weight for boosting rules belonging to the **ambiguity** (AMBIG) category *vs.* the **coverage** (COVER) category.

5 Application to deployed SLU system data

In addition to the previous experiments on benchmark corpora obtained either through a *Wizard-Of-Oz* paradigm (ATIS, MEDIA), or through an automatic process with human supervision (SNIPS, M2M), we decided to test the genericity of our approach on a corpus collected through a deployed service by *Orange* in France.

Orange, the French telco company, has experimented towards the general public the *Djingo* vocal domestic assistant with a set of skills centred on interactions with corporate services (Orange TV, music with its partner Deezer, Orange Radio, telephony), general services (weather, shopping, calendar, news) and general interaction with the speaker (small talks, global commands). According to the customer agreement, and in respect of the French GDPR law, log data have been anonymously

collected and annotated in terms of intents and concept slots. The annotated corpus is built on a weekly basis, and corresponds to a random sub sampling of a whole week logs. The sub-sampling strategy is guided by the annotation capacity for a given week (the average amount of annotations produced by annotators, denoted N_a) and is motivated by the objective of preserving the original distribution of utterances in the test set. Note that the annotations are not produced by crowd sourcing but by expert annotators. Let L be the set of logs gathered during a week, L can be divided into L_s, the subset of already seen utterances, present in the annotation database and L_u=$\overline{L_s}$, the subset of unseen utterances that constitute the pool of candidates for annotation. In a first step, L_u is randomly down-sampled to N_a samples, and the corresponding random sampling probability is applied to L_s in order to derive a down-sampled subset from already annotated samples. The corpus also contains out of domain utterances that are labelled as "NoIntent". The data distribution strategy and the presence of out of domain utterances constitute the most significant differences between this data-set and public benchmark datasets.

Semantic annotations are directly performed on ASR transcriptions and annotated automatic transcriptions are used both for training and testing the NLU model.

For these experiments, the test set is composed of 9984 utterances randomly sub-sampled from a full week of logs. The training corpus is composed of a set of anterior utterances, respecting the usage distribution except that the number of duplicate occurrences for a given utterance is notched to a maximum value of 50 in order to avoid over representation of some very common commands. Overall, the training corpus contains 279375 utterances (with 52132 different utterances). The model ontology is composed of 233 intents and 42 concepts. As can be seen in table 3, the characteristics of the *Djingo* corpus are different from benchmark corpora from several perspectives.

The distribution of utterances reflects the usage and we observe for instance a larger proportion of utterances that are observed in the training corpus, but also a set of out-of-domain utterances and a significant amount of utterances without any concepts.

The SLU model used for this study is a Camembert Transformer [11] fine-tuned on the task of jointly predicting the concept slots with a BIO encoding and the sample's intent, with the intent label set on the [*CLS*] first token, as in the example below.

[CLS]	put	france	info
Set_Radio_Channel	O	B-channel	I-channel

In early experiments we tested different pretrained models and different output layer configurations. As they had similar performances we settle for the fine-tuned Camembert baseline with a simple linear output layer. The model was trained using Pytorch and hyper parameters were chosen using an internal architecture hyper parameter completion toolbox (batch size of 10, learning rate of 5.0e-05, samples padded to a maximum of 50 word pieces, Adam optimizer and 5 epochs).

partition	**AC**	**NCE**	**all**
coverage	86.5%	13.5%	100%
token accuracy	98.6	92.4	97.3
F1 concepts	95.6	83.8	92.2
intent+concepts OK	95.7	79.7	93.5
weight(AMBIG)	-	28.9	-
weight(COVER)	-	71.1	-

Table 10 Evaluation of *easy* (AC) and *difficult* (NCE) partitions of the Djingo corpus thanks to the AC/NCE labels predicted by the complexity classifier

The evaluation of this SLU model on the *Djingo* corpus is given in the last column of table 10. We show 3 metrics: token accuracy, F-measure on concepts and sentence accuracy where a sentence is correct only if both the intent and the concept sequence are correct. As can be seen, the performance are in line with those obtained in table 5

We applied our complexity classifier on the *Djingo* corpus without any retraining or adaptation. We partitioned the corpus into an *easy* set and a *difficult* one according to the label predicted by the classifier. As we can see in table 10, 86.5% of the sentences were labeled as **AC** sentences are 13.5% as **NCE**. By measuring the SLU performance on these 2 subsets, we can check if the AC/NCE prediction are indeed predicting sentence complexity. Results in 10 show that the predicted labels are meaningful since there is a drop of an absolute 16% between results on partition **AC** (95.7) compared to the **NCE** (79.7) partition.

By looking at the distribution of the weights between the *ambiguity* rules and the *coverage* ones, we observed that if issues linked to a lack of coverage in the training date represent 71.1% of the weights, nearly 30% come from ambiguity issues, making this corpus more challenging than ATIS or M2M where a very large majority of rules came from a lack in the training data.

In addition to the use of the AC/NCE prediction, we wanted also to check if the confidence scores given by *Bonzaiboost* on the **NCE** label predictions, could be use to partition further this corpus into sets of different complexity. To this purpose we tested a very simple approach consisting of fixing a threshold δ, then selecting all sentences containing at least one word labeled **NCE** with a score above threshold δ.

By varying δ we obtain the curve of figure 1 which plots the F-measure on concept with respect to the coverage of the corresponding partition. This curve clearly indicates that the NCE label scores are meaningfull as they allow to select sentences of various complexity.

6 Conclusion

We have shown in this study that it was possible to predict sentence complexity without running an SLU system on the data. Just by defining very generic features

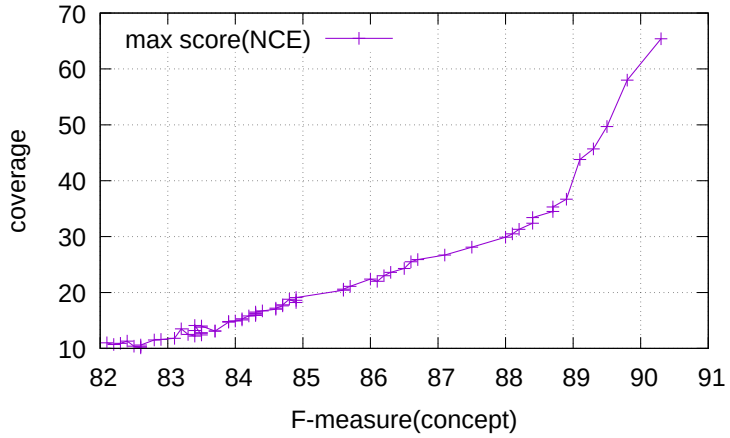

Fig. 1 F-measure vs. coverage for different partitions of the eval corpus according to thresholds applied on the predicted **difficulty** (NCE) score

that could be related either to ambiguity issues, or lack of coverage in the training data, we can process corpora in different languages, topics and semantic models without adaptation. Furthermore the complexity classification model can be analyzed to explain the major complexity factors on the corpus considered, leading to a better characterisation of corpora. Finally, the model was successfully applied on a new corpus collected from a deployed vocal assistant with real-usage distributions, enabling to predict and explain complex utterances.

References

1. Béchet, F., Raymond, C.: Is ATIS too shallow to go deeper for benchmarking Spoken Language Understanding models? In: InterSpeech 2018, pp. 1–5. Hyderabad, India (2018). URL https://hal.inria.fr/hal-01835425
2. Béchet, F., Raymond, C.: Benchmarking benchmarks: introducing new automatic indicators for benchmarking Spoken Language Understanding corpora. In: InterSpeech. Graz, Austria (2019). URL https://hal.archives-ouvertes.fr/hal-02270633
3. Bernier-Colborne, G., Langlais, P.: HardEval: Focusing on challenging tokens to assess robustness of NER. In: Proceedings of the 12th Language Resources and Evaluation Conference, pp. 1704–1711. European Language Resources Association, Marseille, France (2020). URL https://www.aclweb.org/anthology/2020.lrec-1.211
4. Bonneau-Maynard, H., Rosset, S., Ayache, C., Kuhn, A., Mostefa, D.: Semantic Annotation of the French Media Dialog Corpus. In: InterSpeech. Lisbon (2005). URL ftp://tlp.limsi.fr/public/IS052010.PDF
5. Coucke, A., Saade, A., Ball, A., Bluche, T., Caulier, A., Leroy, D., Doumouro, C., Gisselbrecht, T., Caltagirone, F., Lavril, T., Primet, M., Dureau, J.: Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. CoRR **abs/1805.10190** (2018). URL http://arxiv.org/abs/1805.10190

6. Dahl, D.A., Bates, M., Brown, M., Fisher, W., Hunicke-Smith, K., Pallett, D., Pao, C., Rudnicky, A., Shriberg, E.: Expanding the scope of the ATIS task: the ATIS-3 corpus. In: HLT, pp. 43–48 (1994)
7. Fiscus, J.G.: A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (rover). In: 1997 IEEE Workshop on Automatic Speech Recognition and Understanding Proceedings, pp. 347–354. IEEE (1997)
8. Laurent, A., Camelin, N., Raymond, C.: Boosting bonsai trees for efficient features combination : application to speaker role identification. In: Interspeech. Singapour, Singapore (2014). URL https://hal.inria.fr/hal-01025171
9. Lin, H., Lu, Y., Tang, J., Han, X., Sun, L., Wei, Z., Yuan, N.J.: A rigorous study on named entity recognition: Can fine-tuning pretrained model lead to the promised land? In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 7291–7300. Association for Computational Linguistics, Online (2020). DOI 10.18653/v1/2020.emnlp-main.592. URL https://www.aclweb.org/anthology/2020.emnlp-main.592
10. Louvan, S., Magnini, B.: Recent neural methods on slot filling and intent classification for task-oriented dialogue systems: A survey. arXiv preprint arXiv:2011.00564 (2020)
11. Martin, L., Muller, B., Ortiz Suárez, P.J., Dupont, Y., Romary, L., de la Clergerie, É., Seddah, D., Sagot, B.: CamemBERT: a tasty French language model. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 7203–7219. Association for Computational Linguistics, Online (2020). URL https://www.aclweb.org/anthology/2020.acl-main.645
12. Nasr, A., Dary, F., Bechet, F., Favre, B.: Annotation syntaxique automatique de la partie orale du CÉFC. Langages (2020). URL https://hal.archives-ouvertes.fr/hal-02973242
13. Nivre, J., De Marneffe, M.C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C.D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., et al.: Universal dependencies v1: A multilingual treebank collection. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), pp. 1659–1666 (2016)
14. Schapire, R.E., Singer, Y.: Boostexter: A boosting-based system for text categorization. Machine learning **39**(2), 135–168 (2000)
15. Shah, P., Hakkani-Tür, D., Tür, G., Rastogi, A., Bapna, A., Nayak, N., Heck, L.: Building a conversational agent overnight with dialogue self-play. arXiv preprint arXiv:1801.04871 (2018)