Speaker-change Aware CRF for Dialogue Act Classification

Guokan Shang[1,2], **Antoine J.-P. Tixier**[1],
Michalis Vazirgiannis[1,3], **Jean-Pierre Lorré**[2]

[1]École Polytechnique, [2]Linagora, [3]AUEB

Abstract

Recent work in Dialogue Act (DA) classification approaches the task as a sequence labeling problem, using neural network models coupled with a Conditional Random Field (CRF) as the last layer. CRF models the conditional probability of the target DA label sequence given the input utterance sequence. However, the task involves another important input sequence, that of speakers, which is ignored by previous work. To address this limitation, this paper proposes a simple modification of the CRF layer that takes speaker-change into account. Experiments on the SwDA corpus show that our modified CRF layer outperforms the original one, with very wide margins for some DA labels. Further, visualizations demonstrate that our CRF layer can learn meaningful, sophisticated transition patterns between DA label pairs conditioned on speaker-change in an end-to-end way. Code is publicly available[1].

1 Introduction

A conversation can be seen as a sequence of utterances. The task of Dialogue Act (DA) classification aims at assigning to each utterance a DA label to represent its communicative intention. Dialogue acts originate from the notion of *illocutionary force* (speaker's intention in delivering an utterance) introduced back in the theory of Speech Act (Austin, 1962; Searle, 1969). DAs are assigned based on a combination of syntactic, semantic, and pragmatic criteria (Stolcke et al., 2000). As shown in Table 1, some examples of DAs include stating, questioning, answering, etc. The full set of DA labels is predefined. A number of annotation schemes have been developed, varying from domain-specific, such as VERBMOBIL (Alexanderssony et al., 1997), to domain-independent, such as DAMSL (Allen and Core, 1997; Core and Allen, 1997) and DiAML[2] (Bunt et al., 2010; Bunt et al., 2012).

Change	Speaker	Utterance	DA
-	B	Of course I use,	sd
True	A	<laughter>.	x
True	B	credit cards.	+
False	B	I have a couple of credit cards	sd
True	A	Yeah.	b
True	B	and, uh, use them.	+
True	A	Uh-huh,	b
False	A	do you use them a lot?	qy
True	B	Oh, we try not to.	ng

Table 1: Fragment from SwDA conversation sw3332. Statement-non-opinion (sd), Non-verbal (x), Interruption (+), Acknowledge/Backchannel (b), Yes-No-Question (qy), Negative non-no answers (ng).

Automatically detecting DA labels is an essential step towards describing the discourse structure of conversation (Jurafsky et al., 1997). DAs are very useful annotations to a large variety of spoken language understanding tasks, such as utterance clustering (Shang et al., 2019), real-time information retrieval (Meladianos et al., 2017), conversational agents (Higashinaka et al., 2014; Ahmadvand et al., 2019), and summarization (Shang et al., 2018).

It is difficult to predict the DA of a single utterance without having access to the other utterances in the context. For instance, for an utterance such as "`Yeah`", it is hard to tell whether the associated DA should be 'Agreement', 'Yes answer' or 'Backchannel'. Plus, different labels have different transition

GS handled the data, implemented the model, ran the experiments, and generated the plots. GS and AJPT equally participated in the design of the study and the writing of the paper.

[1]`https://bitbucket.org/guokan_shang/da-classification`

[2]accepted to be included in the ISO 24617-2 standard. `https://www.iso.org/standard/76443.html`

Proceedings of the 28th International Conference on Computational Linguistics, pages 450–464
Barcelona, Spain (Online), December 8-13, 2020

probabilities to other labels. E.g., an initial greeting DA is very likely to be followed by another greeting DA. Likewise, a question DA is more likely to be followed by an answer DA. To summarize, it is necessary for a DA classification model to capture dependencies both at the utterance level and at the label level. Recent works (Li and Wu, 2016; Tran et al., 2017; Liu et al., 2017; Kumar et al., 2018; Chen et al., 2018; Raheja and Tetreault, 2019; Li et al., 2019) treat DA classification as a sequence labeling problem. The BiLSTM-CRF model (Huang et al., 2015; Lample et al., 2016), originally introduced for the tasks of POS tagging, chunking and named entity recognition, is the most widely used architecture. In it, a bidirectional recurrent neural network with LSTM cells is first applied to capture the dependencies among consecutive utterances, and then, a Conditional Random Field (CRF) layer is used to capture the dependencies among consecutive DA labels.

CRF is a discriminative probabilistic graphical framework (Koller and Friedman, 2009; Sutton et al., 2012) used to label sequences (Lafferty et al., 2001). It models the conditional probability of a target label sequence given an input sequence. General CRF can essentially model any kind of graphical structure to capture arbitrary dependencies among output variables. For NLP sequence labeling tasks, linear chain CRF is the most common variant. The labels are arranged in a linear chain, i.e., only neighboring labels are dependent (first-order Markov assumption). The BiLSTM-CRF architecture employs a linear chain CRF. Hence, for brevity, in the rest of this paper, the term CRF is short for linear chain CRF.

Recently, neural versions of the CRF have been developed mainly for NLP sequence labeling tasks (Collobert et al., 2011; Huang et al., 2015; Lample et al., 2016). While traditional CRF requires defining a potentially large set of handcrafted feature functions (each weighted with a parameter to be trained), neural CRF has only two parameterized feature functions (emission and transition) that are trained with the other parameters of the network in an end-to-end fashion.

2 Motivation

Most sequence labeling tasks in NLP, such as POS tagging, chunking, and named entity recognition, involve only two sequences: input and target. In DA classification however, we have access to an additional input sequence, that of speaker-identifiers. This extra input could, in principle, greatly improve DA prediction. Indeed, research on turn management (Sacks et al., 1974) has shown that dialogue participants do not start or stop speaking arbitrarily, but follow an underlying turn-taking system to occupy or release the speaker role (Petukhova and Bunt, 2009). For instance, the last two utterances in Table 1 illustrate a non-arbitrary change of speakers, following a turn-allocation action (here, a question). In this conversational situation, speaker B has to take the turn, to respond to speaker A. To sum up, the sequences of DAs and speakers are tightly interconnected.

However, state-of-the-art DA classification models ignore the sequence of speaker-identifiers (Kumar et al., 2018; Chen et al., 2018; Raheja and Tetreault, 2019; Li et al., 2019). This is a clear limitation. To address this limitation, we propose in this paper a simple modification of the CRF layer where the label transition matrix is conditioned on speaker-change. We evaluate our modified CRF layer within the BiLSTM-CRF architecture, and find that on the SwDA corpus, it improves performance compared to the original CRF. Furthermore, visualizations demonstrate that sophisticated transition patterns between DA label pairs, conditioned on speaker-change, can be learned in an end-to-end way.

3 Related work

In this section, we first introduce the two major DA classification approaches, and then focus on previous work involving the use of BiLSTM-CRF and speaker information.

Multi-class classification. In this first approach, consecutive DA labels are considered to be independent. The DA label of each utterance is predicted in isolation by a classifier such as, e.g., naive Bayes (Grau et al., 2004), Maxent (Venkataraman et al., 2005; Ang et al., 2005), or SVM (Liu, 2006). Since the first application of neural networks to DA classification by Ries (1999), deep learning has shown promising results even with some simple architectures (Khanpour et al., 2016; Shen and Lee, 2016). More recent work developed more advanced models, and started taking into account the dependencies among consecutive utterances (Kalchbrenner and Blunsom, 2013; Lee and Dernoncourt, 2016; Ortega and Vu, 2017; Bothe et

al., 2018). For example, in Bothe et al. (2018), the representations of the current utterance and the three preceding utterances are fed into a RNN, and the last annotation is used to predict the DA label of the current utterance.

Sequence labeling. In the second approach, the DA labels for all the utterances in the conversation are classified together. Traditional work uses statistical approaches such as Hidden Markov Models (HMM) (Stolcke et al., 2000; Surendran and Levow, 2006; Tavafi et al., 2013) and CRFs (Lendvai and Geertzen, 2007; Zimmermann, 2009; Kim et al., 2010) with handcrafted features. In HMM based approaches, the DA labels are hidden states and utterances are observations emanating from these states. The hidden states are evolving according to a discourse grammar, which essentially is an n-gram language model trained on DA label sequences. Following advances in deep learning, neural sequence labeling architectures (Huang et al., 2015; Reimers and Gurevych, 2017; Yang et al., 2018; Cui and Zhang, 2019) have set new state-of-the-art performance. Two major architectures have been tested: BiLSTM-Softmax (Li and Wu, 2016; Tran et al., 2017; Liu et al., 2017) and BiLSTM-CRF. This study focuses on the BiLSTM-CRF architecture.

BiLSTM-CRF. Kumar et al. (2018) were the first to introduce the BiLSTM-CRF architecture for DA classification. Their model is hierarchical and consists of two levels, where at level 1, the text of each utterance is separately encoded by a shared bidirectional LSTM (BiLSTM) with last-pooling, resulting in a sequence of vectors. At level 2, that sequence is passed through another BiLSTM topped by a CRF layer. At test time, the optimal output label sequence is retrieved from the trained model by Viterbi algorithm (Viterbi, 1967). Chen et al. (2018) and Raheja and Tetreault (2019) improved on the previous model by adding different attention mechanisms. Li et al. (2019) discovered that performing topic classification as an auxiliary task, can assist in predicting DA labels. The topic of each utterance is automatically determined using Latent Dirichlet Allocation (Blei et al., 2003). Their model consists of two BiLSTM-CRF architectures for predicting simultaneously the target DA label sequence and the target topic label sequence. This model represents the state-of-the-art in DA classification.

Speaker information. There are only a few previous works that consider the sequence of speaker-identifiers for DA classification. In Bothe et al. (2018), the utterance representation is the concatenation of the one-hot encoded speaker-identifier, e.g., A as [1, 0] and B as [0, 1], with the output of the RNN-based character-level utterance encoder. By contrast, Li and Wu (2016) and Liu et al. (2017) choose to concatenate the speaker-change vector with the representation obtained via their CNN-based and RNN-based word-level utterance encoders. Speaker-change is binary as shown in Table 1, obtained by checking if the current utterance is from the same or different speaker as the previous one. Venkataraman et al. (2005) also include speaker-change as one of the handcrafted features for their Maxent classifier.

Apart from the naive concatenation approaches described above, Kalchbrenner and Blunsom (2013) proposed to let the recurrent and output weights of the RNN cell be conditioned on speaker-identifier, i.e., a speaker-aware RNN cell. Stolcke et al. (2000) proposed to train different discourse grammars for different speakers, to guide DA label transitions in HMM.

4 Model

Here, we describe the general BiLSTM-CRF model for DA classification, shown in Fig. 1. Then, in the next section, we present our modification of the CRF layer that takes speaker-change into account.

Notation. Let us denote by $\{(\mathbf{x}^t, y^t)\}_{t=1}^T$ a conversation of length T. $X = \{\mathbf{x}^t\}_{t=1}^T$ is the sequence of utterances, where each utterance $\mathbf{x}^t = \{x_n^t\}_{n=1}^N$ is itself a sequence of words of length N. $Y = \{y^t\}_{t=1}^T$ denotes the target sequence, where $y^t \in \mathcal{Y}$ is the set of all possible DA labels of size $|\mathcal{Y}| = K$. We use y^t to denote the label and its integer index interchangeably.

Utterance encoder. Each utterance is separately encoded by a shared forward RNN with LSTM cells. Only the last annotation $\mathbf{u}_N^t$ is retained (last pooling). We are left with a sequence of utterance embeddings $\{\mathbf{u}^t\}_{t=1}^T$.

BiLSTM layer. The sequence of utterance embeddings $\{\mathbf{u}^t\}_{t=1}^T$ is then passed on to a bidirectional LSTM, returning the sequence of conversation-level utterance representations $\{\mathbf{v}^t\}_{t=1}^T$.

CRF layer. $\mathbf{v}^t$ can already be used to predict locally the label at each time step in isolation, through a

dense layer with softmax activation, which results in the BiLSTM-Softmax architecture. However this might lead to a non-optimal global solution, if we consider the output DA label sequence as a whole.

On the other hand, CRF models the conditional probability $P(Y|X)$ of an entire target sequence Y given an entire input sequence X. Thus, it guarantees an optimal global solution, under the first order Markov assumption. More precisely:

$$P(Y|X) = \frac{\exp(\psi(X,Y))}{\sum_{\tilde{Y}} \exp(\psi(X,\tilde{Y}))} \quad (1)$$

where $\psi(X,Y)$ is a feature function that assigns a *path score* to the label sequence Y, giving X. Then, a softmax function is used to yield the conditional probability, where $\tilde{Y}$ denotes one of all possible label sequences (paths).

$\psi(X,Y)$ is defined as the sum of *emission scores* (or state scores) and *transition scores* over all time steps (Morris and Fosler-Lussier, 2006; Chen and Moschitti, 2019):

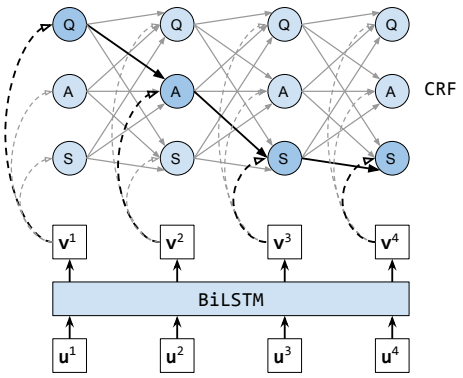

Figure 1: BiLSTM-CRF, for an example with $\{\mathbf{u}^t\}_{t=1}^{T=4}$ (utterance embeddings) as input and Q, A, S, S (DA labels) as target. Three possible labels $\{Q, A, S\}$ stand for Question, Answer, and Statement, respectively.

$$\psi(X,Y) = \sum_{t=1}^{T} h(y^t, X) + \sum_{t=1}^{T-1} g(y^t, y^{t+1}) \quad (2)$$

Emission (state) scores are assigned to the dashed top-down edges (nodes) in Fig. 1, computed as follows:

$$h(y^t, X) = (\mathbf{W}\mathbf{v}^t + \mathbf{b})[y^t] \quad (3)$$

where the conversation-level utterance representation $\mathbf{v}^t$ is converted into a vector of size K and $[y^t]$ denotes the element at index y^t. Higher values of $h(y^t, X)$ indicate that the model is more confident in predicting the output label y^t at time step t.
Transition scores are assigned to the solid left-to-right edges in Fig. 1, computed as follows:

$$g(y^t, y^{t+1}) = \mathbf{G}[y^t, y^{t+1}] \quad (4)$$

where $\mathbf{G}$ is the label transition matrix of size $K \times K$. E.g, the element $\mathbf{G}[y^t, y^{t+1}]$ is the transition score from label y^t to label y^{t+1}. Note that the transition matrix is shared across all time steps.

The CRF layer is parameterized by $\mathbf{W}$, $\mathbf{b}$, and $\mathbf{G}$. To learn these parameters and those of the previous layers, maximum likelihood estimation is used. For a training set of M conversations, the loss can be written as:

$$\mathcal{L} = \sum_{m=1}^{M} -\log P(Y^m|X^m) \quad (5)$$

At test time, the optimal output label sequence, i.e., $Y^* = \text{argmax}_{\tilde{Y}} P(\tilde{Y}|X)$ for unseen X, is obtained with the Viterbi decoding algorithm (Viterbi, 1967). Due to the Markov property of the linear chain CRF, the computations of Viterbi algorithm and the normalization term in Eq. 1 can be broken down into a series of sub-problems over time in a recursive manner, which are solved via dynamic programming (Bellman, 1966), with polynomial complexity $\mathcal{O}(TK^2)$.

5 Our contribution

Given the sequence of speaker-identifiers $S = \{s^t\}_{t=1}^{T}$, we can instantly derive the sequence of speaker-changes $Z = \{z^{t,t+1}\}_{t=1}^{T-1}$ by comparing neighbors. E.g., $z^{2,3} = 0$ means the speaker does not change from time $t = 2$ to $t = 3$.

We extend the original CRF so that it considers as additional input, the sequence Z. That is, CRF now models $P(Y|X,Z)$ instead of just $P(Y|X)$. In other words, the prediction of the DA label sequence is now conditioned both on the utterance sequence and the speaker-change sequence. Specifically, transition scores in our modified CRF layer are computed as follows:

$$g(y^t, y^{t+1}, z^{t,t+1}) = (1 - z^{t,t+1}) * \mathbf{G}_0[y^t, y^{t+1}] + z^{t,t+1} * \mathbf{G}_1[y^t, y^{t+1}] \quad (6)$$

where $\mathbf{G}_0$ and $\mathbf{G}_1$ are label transition matrices of size $K \times K$, corresponding respectively to the "speaker unchanged" and "speaker changed" cases.

6 Experimental setup

Dataset. We experiment on the widely-used SwDA[3] (Switchboard Dialogue Act) dataset (Jurafsky et al., 1997; Stolcke et al., 2000). This corpus contains telephone conversations recorded between two randomly selected speakers talking about one of various general topics (air pollution, music, football, etc.). In this dataset, utterances are annotated with 42 mutually exclusive DA labels, based on the SWBD-DAMSL annotation scheme (Jurafsky et al., 1997). Inter-annotator agreement is 84%. The frequency of the 10 most represented DA labels are illustrated in Fig. 2. We can see that labels are highly imbalanced and follow a long-tailed distribution. Detailed statistics for all 42 labels are provided in Appendix B.

We adopt the same training, validation and testing partition as previous work (Lee and Dernoncourt, 2016)[4], consisting of 1003, 112, and 19 conversations, respectively.

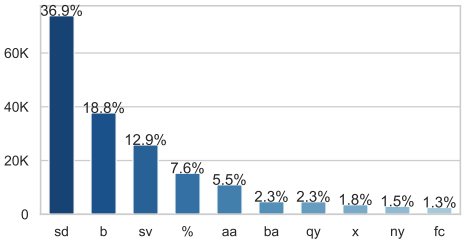

Figure 2: Counts and frequencies of the 10 most represented DA labels in the SwDA dataset. There are 200444 utterances in total.

A note about the '+' tag. The '+' tag, as shown in Table 1, accounts for 8.1% of the total annotations, but is not part of the default label set. That tag is used to mark the remaining parts of an utterance that has been interrupted by the other speaker. While most of the previous works did not predict, or even mention this tag, some efforts considered it as a 43^{rd} DA label and predicted it (Lee and Dernoncourt, 2016; Raheja and Tetreault, 2019).

In this paper, we followed the approach of (Webb et al., 2005; Milajevs and Purver, 2014; Kim et al., 2017), and used the '+' tag to reconnect, bottom-up, all the parts of an interrupted utterance together. E.g., in Table 1, the parts "`Of course I use,`" and "`credit cards.`", uttered by speaker B, are reconnected into "`Of course I use, credit cards.`", which becomes the new first utterance. It is followed by "`<laughter>`", uttered by speaker A. We opted for this approach as predicting the DA of a broken utterance sometimes does not make sense. For instance, in this situation with three utterances: (1) "`A: so, (Wh-Question)`", (2) "`B: <throat_clearing> (Non-verbal)`", and (3) "`A: what's your name? (+)`", it is very difficult to correctly predict that utterance 1 is a question. And predicting anything other than a question-related tag for utterance 3 does not really make sense. Reconstructing 1 and 3 into a single utterance "`A: so, what's your name? (Wh-Question)`" solves both issues.

Implementation and training details. Disfluency markers (Meteer et al., 1995) were filtered out and all characters converted to lowercase. We used some optimal hyperparameters provided by Kumar et al. (2018). E.g., 0.2 dropout was applied to the utterance embeddings and conversation-level utterance representations, and all LSTM layers had 300 hidden units. The embedding layer was initialized using 300-dimensional word vectors pre-trained with the gensim (Řehůřek and Sojka, 2010) implementation of `word2vec` (Mikolov et al., 2013) on the utterances of the training set, and was frozen during training. Vocabulary size was around 21K, and out-of-vocabulary words were mapped to a special token [UNK], randomly initialized.

Models were trained with the Adam optimizer (Kingma and Ba, 2015). Early stopping was used on the validation set with a patience of 5 epochs and a maximum number of epochs of 100. The best epoch was selected as the one associated with the highest validation accuracy. Usually, the best epoch was within the first 10. We set our batch-size to be 1, i.e, one conversation for one training step. Batch sizes of 1, 2, 4, 8, and 16 were also tried, without observing significant differences.

[3] `https://github.com/cgpotts/swda`

[4] `https://github.com/Franck-Dernoncourt/naacl2016`

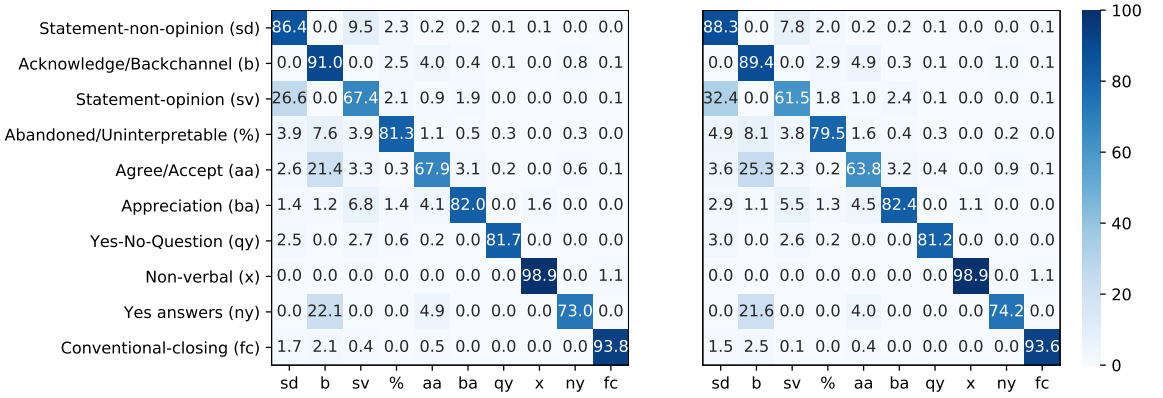

Figure 3: Normalized confusion matrices, averaged over 10 runs, for the 10 most frequent DA labels (90.9% of all annotations). Left: our model, right: base model. Rows (columns) correspond to true (predicted) classes.

7 Quantitative results

Performance comparison. Table 2 reports the results in terms of classification accuracy, averaged over 10 runs to account for the randomness of SGD. Model a) uses our modified CRF layer. Model b) has exactly the same architecture as a), but uses a vanilla CRF layer.

Results show that, in terms of overall accuracy on the test set of 42 DA labels, our model a) outperforms the base model b) by 1%. Moreover, the small standard deviations highlight the consistency of this improvement over the 10 runs. Note that this performance gain is solely caused by our modified CRF layer capturing speaker-change, and is greater than the gains of 0.26% (Liu et al., 2017) and 0.09% (Bothe et al., 2018) reported by previous attempts at leveraging speaker information.

To interpret the results in more detail, we show in Fig. 3 the confusion matrices of our model and the base model, for the 10 most frequent DA labels, representing close to 91% of all annotations. The rows correspond to true classes, and the columns to predicted classes.[5] By looking at the diagonals, we can see that our model (on the left) better predicts 6 labels out of 10 with absolute accuracy gains of up to 5.9% (for statement-opinion, `sv`)[6] and is on par with the baseline model for one label (non-verbal, `x`), at 98.9%. By looking at off-diagonal values, miss rates are decreased up to 5.8% (for `sv` misclassified as `sd`) by our model. Also, our model provides a large boost for the (Acknowledge/Backchannel, Agree/Accept), or (`b`, `aa`) pair. It increases the respective accuracies by 1.6% (89.4%→91.0%) and 4.1% (63.8%→67.9%). The respective miss rates are decreased by 0.9% (4.9%→4.0%) and 3.9% (25.3%→21.4%), respectively for `b` misclassified as `aa` and `aa` misclassified as `b`. This is to be noted, as these two labels are among the most frequently confused pairs (Kumar et al., 2018; Chen et al., 2018).

Although our model achieves significant gains on a majority of the most frequent labels, it decreases performance for the most frequent label, `sd`, which accounts for 36.9% of all labels, as shown by Fig. 2. This explains why, in terms of overall accuracy, our improvements are modest. In addition, the performance drop regarding `sd` can be interpreted as a consequence of the trade-off between `sd` and `sv`, since the distinction between them was very hard to make even by annotators (Jurafsky et al., 1997). This can be demonstrated in terms of precision, recall, and F1 score, as shown in Table 3. We can observe that, as opposed to the base model, our model has lower `sd` and higher `sv` recall values. A similar observation can be made for precision scores. Thus, the prediction between `sd` and `sv` is a trade-off made by models. It is also interesting to note that our model is superior for both labels in terms of F1 score.

We can observe in Table 4 that our model brings improvement where it is most necessary, i.e., for the most difficult and rare DAs (20%). Full details are provided in Appendix A, along with the corresponding

[5]Note that our confusion matrices were row-wise normalized by class size. So we use the terms accuracy (per class) to denote diagonal values (equivalent to recall or hit rate), and miss rate for off-diagonal values.

[6]The margins are even larger (up to 20%) on some less frequent labels, as shown by the results in Appendix A.

Model		BiLSTM input	CRF extra input	Accuracy (% ± SD)
a)	Our CRF	$\mathbf{u}^t$	SC	**78.70** ± .37
a1)		$\mathbf{u}^t$ + SI	SC	78.32 ± .28
a2)		$\mathbf{u}^t$ + SC	SC	78.65 ± .47
b)	Vanilla CRF	$\mathbf{u}^t$	-	77.69 ± .38
b1)		$\mathbf{u}^t$ + SI	-	77.86 ± .61
b2)		$\mathbf{u}^t$ + SC	-	78.33 ± .71
c)	Softmax	$\mathbf{u}^t$	-	77.80 ± .48
c1)		$\mathbf{u}^t$ + SI	-	77.73 ± .44
c2)		$\mathbf{u}^t$ + SC	-	78.33 ± .49
a) + b) ensembling		$\mathbf{u}^t$	SC	**78.89** ± .20
a) + b) joint training		$\mathbf{u}^t$	SC	78.27 ± .47

Table 2: Results, averaged over 10 runs. SI: speaker-identifier, SC: speaker-change, $\mathbf{u}^t$: utterance embedding, ±: standard deviation.

		P	R	F1
Our	sd	80.49	86.36	83.32
	sv	71.54	67.41	69.42
Vanilla	sd	77.83	88.32	82.74
	sv	73.24	61.48	66.84

Table 3: Precison, Recall, and F1 score (%) of our model vs. base model on the `sd` and `sv` labels.

	Ours	Vanilla	Diff.
10 best DAs	37.08	31.70	+ 5.38
10 worst DAs	59.67	64.54	- 4.87

Table 4: Average accuracy (%) of our model vs. base model on the 10 DAs best and worst predicted by our model (resp. representing 20% and 40% of all annotations).

confusion matrices.

The benefits of considering speaker information vary across DA labels. Our model and the base model performed very closely on 4 labels: Non-verbal (`x`), Conventional-closing (`fc`), Appreciation (`ba`), and Yes-No-Question (`qy`). We found that the utterances of these labels contain clear lexical cues that can be mapped to corresponding DA labels in a *non-ambiguous* way. Some examples include "`<laughter>`" → `x`, "`Bye-bye.`" → `fc`, "`That's great.`" → `ba`, and "`Do you ...?`" → `qy`. In other words, predicting well these four DAs does not require having access to speaker information. It can be done solely from the text of the current utterance. Having access to context is not even required. This explain why our speaker-aware CRF is not helpful here.

This interpretation is supported by the fact that, as explained in Appendix A, our model is most useful for the DAs that require speaker-change awareness.

Ensembling and joint training. Since the model using our CRF and the model using the vanilla CRF appear to have their own strengths and weaknesses, we tried combining them to improve performance. More precisely, we experimented with two approaches. First, an ensembling approach that combines the predictions of the two trained models by averaging their emission and transition scores (respectively). Second, a joint training approach that combines the two models into a new one and trains it from scratch. In that second model, our CRF and the vanilla CRF are combined, and transition scores are computed as:

$$g(y^t, y^{t+1}, z^{t,t+1}) = \mathbf{G}_{basis}[y^t, y^{t+1}] + (1 - z^{t,t+1}) * \mathbf{G}_0[y^t, y^{t+1}] + z^{t,t+1} * \mathbf{G}_1[y^t, y^{t+1}] \quad (7)$$

where G_{basis} is the transition matrix as in the original CRF, used at each time step, while $\mathbf{G}_0$/$\mathbf{G}_1$ are applied only when the speaker does not change/changes, as in our modified CRF layer.

Results in Table 2 show that the ensemble model reaches new best performance at 78.89, providing close to a 0.2 boost from our model, and a 1.2 boost from the vanilla CRF model. On the other hand, the jointly-trained model does not outperform our model. After inspecting the transition matrices for the two cases ($\mathbf{G}_{basis} + \mathbf{G}_0$) and ($\mathbf{G}_{basis} + \mathbf{G}_1$), we found that the addition of $\mathbf{G}_{basis}$ blurred the label transition patterns.

Ablation studies. Our results showed that considering speaker information improves DA classification. Then, we wanted to confirm whether our way of taking speaker information into account (at the CRF level) was the most effective. To this purpose, we trained two other base models, both using the vanilla CRF. These two models respectively concatenate the one-hot encoded speaker-identifier vector (noted SI, of size 2) and the binary speaker-change vector (noted SC, of size 1) with the utterance embedding $\mathbf{u}^t$.[7] Results, shown in rows b1 and b2 of Table 2, show that while they bring improvement compared to the

[7] proposed in Liu et al. (2017) and Bothe et al. (2018), but not in the context of BiLSTM-CRF.

basic base model (row b), these two approaches are not able to yield as big of a gain as the model using our modified CRF layer, indicating that taking speaker-change into account at the CRF level is superior.

For the sake of completeness, we repeated these experiments with our model. Results, available in rows a1 and a2 of Table 2, show that performance was not improved (78.32 and 78.65 vs. 78.70). Thus, it seems that taking speaker information into account twice, both at the BiLSTM level and at the CRF level, is not useful, or at least, not in this way.

Results in Table 2 also show that SC is a better feature than SI in general.

BiLSTM-CRF vs. BiLSTM-Softmax. To the best of our knowledge, no previous study has compared BiLSTM-CRF to BiLSTM-Softmax on the DA classification task. Hence, in this paper, we decided to compare between these two models. Results reveal that the models using BiLSTM-Softmax (rows c, c1, and c2) are competitive with the ones using BiLSTM-CRF (rows b, b1, and b2). More specifically, BiLSTM-Softmax outperforms BiLSTM-CRF with text features only (rows b vs. c), by a slight 0.11 margin, but it is the opposite for text + SI (b1 vs. c1, 0.13 difference). With text + SC (b2 vs. c2), they achieve similar performance.

These results are not very surprising, since, on other tasks than DA classification, multiple recent works have reported that BiLSTM-CRF does not always outperform BiLSTM-Softmax (Reimers and Gurevych, 2017; Yang et al., 2018; Cui and Zhang, 2019). For example, in Yang et al. (2018), CRF brought improvement for named entity recognition and chunking, but not for POS tagging. One of the reasons might be that the simple Markov label transition model of CRF does not give much information gain over strong neural encoding (Cui and Zhang, 2019). That is, BiLSTM may be expressive enough to implicitly capture the *obvious* dependencies among labels.

In any case, the model equipped with our CRF layer (row a) outperforms all variants of BiLSTM-Softmax and BiLSTM-vanilla_CRF. This suggests that our CRF layer can capture richer and *not obvious* label dependencies given speaker information, which, in the end, makes the use of a CRF layer valuable in assisting DA classification.

8 Qualitative results

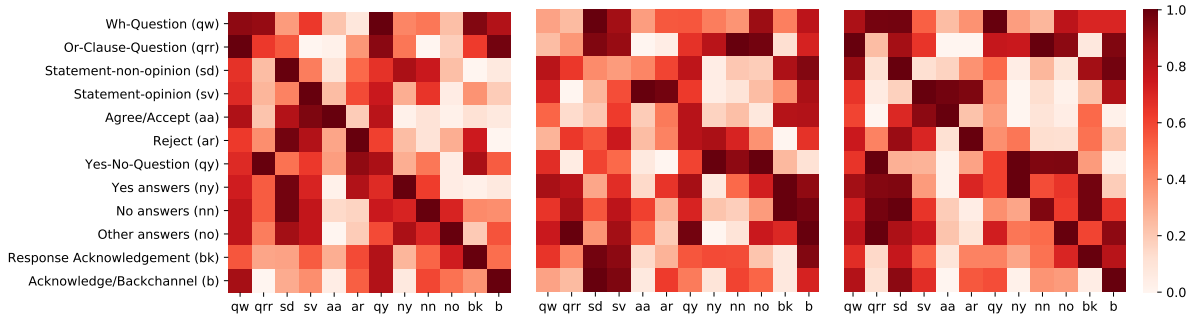

Figure 4: Normalized transition matrices (averaged over 10 runs). Left and center: $\mathbf{G}_0$ (speaker unchanged) and $\mathbf{G}_1$ (speaker changed) of our CRF layer. Right: $\mathbf{G}$ of vanilla CRF layer. The darker, the greater the score.

Visualization of transition matrices. We illustrate, in Fig. 4, the transition matrices $\mathbf{G}_0$ and $\mathbf{G}_1$ of our CRF layer, together with the single matrix $\mathbf{G}$ of the vanilla CRF layer. This visualization is done for 12 labels that are easy to interpret, such as statements, questions, answers, etc. We can observe some interesting patterns, sometimes matching intuition, and sometimes harder to interpret. We report some of the most interesting findings below:

1) Overall, $\mathbf{G}_0$ and $\mathbf{G}_1$ are not identical, which means that different transition patterns are associated with the "speaker unchanged" and "speaker changed" cases. The dark diagonal of $\mathbf{G}_0$ shows that when the speaker does not change, the majority of labels tend to carry over to the next utterance. On the opposite, $\mathbf{G}_1$ clearly shows that changing speakers very often induce a change in DA.

2) questions starting with words including: 'what', 'how', etc. (qw label) tend to transfer to statements (sd and sv) and to other answers (no, e.g., "I don't know") if the speaker changed, but to other forms of questions, yes-no questions and questions starting with the word 'or' (qy and qrr), or to acknowledgements (bk and b) if the speaker did not change. This probably corresponds to instances when the same speaker clarifies, elaborates on, or answers, an original question.
3) sv label (statement with opinion) tends to transition to Agree/Accept aa and Reject ar if the speaker changed, while no such clear pattern can be observed for the sd label (statement without opinion).
4) qy label (yes/no questions) tend to transfer to answer labels ny (yes), nn (no), no (other) if the speaker changed, and to another type of question (e.g., or-clause) if the speaker did not change. Again, the latter surely corresponds to the case where a given speaker elaborates on his or her original question.
5) answer labels (ny, nn, no) tend to be followed by Response Acknowledgement bk and Acknowledge/Backchannel b if the speaker changed, but by themselves or statements (sd and sv) if the speaker did not change.

As far as the transition matrix $\mathbf{G}$ of the vanilla CRF layer (right of Fig. 4), we can observe that it tries to capture, at the same time, the transition patterns of both the "speaker changed" and "speaker unchanged" cases. For example, sv equally tends to transfer to sv, aa and ar in $\mathbf{G}$, while the transitions towards sv/aa,ar are only probable if the speaker stays the same/changes, as clearly illustrated by $\mathbf{G_0}$/$\mathbf{G_1}$. Obviously, using two matrices as in our approach gives much more expressiveness to the model in capturing DA label transition patterns. To summarize, visualizations show that the transition matrices $\mathbf{G}_0$ and $\mathbf{G}_1$ in our modified CRF layer are able to encode speaker-change-aware, sophisticated DA transition patterns.

9 Discussion

Note that, for our utterance encoder, we also experimented with a bidirectional LSTM (also with last pooling), as in Kumar et al. (2018), and with a bidirectional LSTM with self-attention mechanism (Yang et al., 2016). However, since they were not giving better results, we opted for the simplest option. One possible explanation for the self-attention mechanism not being helpful could be the very short size of the utterances in the SwDA dataset (68.7% of utterances are shorter than 10 tokens). On such short sequences, a RNN with a 300-dimensional hidden layer is very likely able to keep the full sequence into memory. As far as why a forward RNN suffices, it should be noted that with last pooling, the last time step corresponds to the first annotation of the backward RNN. This is not adding much information to the last annotation of the forward RNN, which represents the entire sequence.

Our goal was not to exceed the state-of-the-art accuracy reported in Li et al. (2019) and Raheja and Tetreault (2019), this is why we used simple models in all of our experiments. However, our improved CRF layer can be directly plugged into more advanced architectures, such as Att-BiLSTM-CRF (Luo et al., 2018) or Transformer-CRF (Chen et al., 2019; Zhang and Wang, 2019; Yan et al., 2019; Winata et al., 2019), and should in principle be able to boost performance regardless of the model used.

Future research should be devoted to address the limitation of the Markov property of CRF layer, by developing a model that is capable of capturing longer-range dependencies within and among the three sequences: that of speakers, utterances, and DA labels.

10 Conclusion

In this paper, we focused on demonstrating that taking speaker information into consideration was beneficial to the task of DA classification, with the BiLSTM-CRF architecture. We proposed a modified CRF layer that takes as extra input the sequence of speaker-changes. Experiments conducted on the SwDA dataset showed that our CRF layer outperforms vanilla CRF, and brings greater gains than previous attempts at taking speaker information into account. Moreover, visualizations confirmed that our improved CRF was able to learn complex speaker-change aware DA transition patterns in an end-to-end way.

Acknowledgments

This work was supported by the LinTO project (Lorré et al., 2019).

References

Ali Ahmadvand, Jason Ingyu Choi, and Eugene Agichtein. 2019. Contextual dialogue act classification for open-domain conversational agents. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1273–1276.

Jan Alexanderssony, Bianka Buschbeck-Wolfz, Tsutomu Fujinamiz, Elisabeth Maiery, Norbert Reithingery, Birte Schmitzx, and Melanie Siegelyy. 1997. Dialogue acts in verbmobil-2. *DFKI*.

James Allen and Mark Core. 1997. Draft of damsl: Dialog act markup in several layers.

Jeremy Ang, Yang Liu, and Elizabeth Shriberg. 2005. Automatic dialog act segmentation and classification in multiparty meetings. In *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, volume 1, pages I–1061. IEEE.

John Langshaw Austin. 1962. *How to do things with words*, volume 88. Oxford university press.

Richard Bellman. 1966. Dynamic programming. *Science*, 153(3731):34–37.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Chandrakant Bothe, Cornelius Weber, Sven Magg, and Stefan Wermter. 2018. A context-based approach for dialogue act recognition using simple recurrent neural networks. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May. European Language Resources Association (ELRA).

Harry Bunt, Jan Alexandersson, Jean Carletta, Jae-Woong Choe, Alex Chengyu Fang, Koiti Hasida, Kiyong Lee, Volha Petukhova, Andrei Popescu-Belis, Laurent Romary, Claudia Soria, and David Traum. 2010. Towards an ISO standard for dialogue act annotation. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May. European Language Resources Association (ELRA).

Harry Bunt, Jan Alexandersson, Jae-Woong Choe, Alex Chengyu Fang, Koiti Hasida, Volha Petukhova, Andrei Popescu-Belis, and David Traum. 2012. ISO 24617-2: A semantically-based standard for dialogue annotation. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 430–437, Istanbul, Turkey, May. European Language Resources Association (ELRA).

Lingzhen Chen and Alessandro Moschitti. 2019. Transfer learning for sequence labeling using source model and target data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6260–6267.

Zheqian Chen, Rongqin Yang, Zhou Zhao, Deng Cai, and Xiaofei He. 2018. Dialogue act recognition via crf-attentive structured network. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, page 225–234, New York, NY, USA. Association for Computing Machinery.

Qian Chen, Zhu Zhuo, and Wen Wang. 2019. Bert for joint intent classification and slot filling. *arXiv preprint arXiv:1902.10909*.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12(Aug):2493–2537.

Mark G Core and James Allen. 1997. Coding dialogs with the damsl annotation scheme. In *AAAI fall symposium on communicative action in humans and machines*, volume 56, pages 28–35. Boston, MA.

Leyang Cui and Yue Zhang. 2019. Hierarchically-refined label attention network for sequence labeling. *arXiv preprint arXiv:1908.08676*.

Sergio Grau, Emilio Sanchis, Maria Jose Castro, and David Vilar. 2004. Dialogue act classification using a bayesian approach. In *9th Conference Speech and Computer*.

Ryuichiro Higashinaka, Kenji Imamura, Toyomi Meguro, Chiaki Miyazaki, Nozomi Kobayashi, Hiroaki Sugiyama, Toru Hirano, Toshiro Makino, and Yoshihiro Matsuo. 2014. Towards an open-domain conversational system fully based on natural language processing. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 928–939, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Dan Jurafsky, Liz Shriberg, and Debra Biasca. 1997. Switchboard swbd-damsl shallow-discourse-function annotation coders manual. *Institute of Cognitive Science Technical Report*.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent convolutional neural networks for discourse compositionality. In *Proceedings of the Workshop on Continuous Vector Space Models and their Compositionality*, pages 119–126, Sofia, Bulgaria, August. Association for Computational Linguistics.

Hamed Khanpour, Nishitha Guntakandla, and Rodney Nielsen. 2016. Dialogue act classification in domain-independent conversations using a deep recurrent neural network. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2012–2021, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Su Nam Kim, Lawrence Cavedon, and Timothy Baldwin. 2010. Classifying dialogue acts in one-on-one live chats. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 862–871, Cambridge, MA, October. Association for Computational Linguistics.

Geonmin Kim, Hwaran Lee, Bokyeong Kim, and Soo-young Lee. 2017. Compositional sentence representation from character within large context text. In Derong Liu, Shengli Xie, Yuanqing Li, Dongbin Zhao, and El-Sayed M. El-Alfy, editors, *Neural Information Processing*, pages 674–685, Cham. Springer International Publishing.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Daphne Koller and Nir Friedman. 2009. *Probabilistic graphical models: principles and techniques*. MIT press.

Harshit Kumar, Arvind Agarwal, Riddhiman Dasgupta, and Sachindra Joshi. 2018. Dialogue act sequence labeling using hierarchical encoder with crf. *AAAI Conference on Artificial Intelligence*.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270, San Diego, California, June. Association for Computational Linguistics.

Ji Young Lee and Franck Dernoncourt. 2016. Sequential short-text classification with recurrent and convolutional neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 515–520. Association for Computational Linguistics.

Piroska Lendvai and Jeroen Geertzen. 2007. Token-based chunking of turn-internal dialogue act sequences. In *Proceedings of the 8th SIGDIAL Workshop on Discourse and Dialogue*, pages 174–181.

Wei Li and Yunfang Wu. 2016. Multi-level gated recurrent neural network for dialog act classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1970–1979, Osaka, Japan, December. The COLING 2016 Organizing Committee.

Ruizhe Li, Chenghua Lin, Matthew Collinson, Xiao Li, and Guanyi Chen. 2019. A dual-attention hierarchical recurrent neural network for dialogue act classification. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 383–392, Hong Kong, China, November. Association for Computational Linguistics.

Yang Liu, Kun Han, Zhao Tan, and Yun Lei. 2017. Using context information for dialog act classification in DNN framework. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2170–2178, Copenhagen, Denmark, September. Association for Computational Linguistics.

Yang Liu. 2006. Using svm and error-correcting codes for multiclass dialog act classification in meeting corpus. In *Ninth International Conference on Spoken Language Processing*.

Jean-Pierre Lorré, Isabelle Ferrané, Francisco Madrigal, Michalis Vazirgiannis, and Christophe Bourguignat. 2019. Linto: Assistant vocal open-source respectueux des données personnelles pour les réunions d'entreprise. *APIA*, page 63.

Ling Luo, Zhihao Yang, Pei Yang, Yin Zhang, Lei Wang, Hongfei Lin, and Jian Wang. 2018. An attention-based bilstm-crf approach to document-level chemical named entity recognition. *Bioinformatics*, 34(8):1381–1388.

Polykarpos Meladianos, Antoine Tixier, Ioannis Nikolentzos, and Michalis Vazirgiannis. 2017. Real-time keyword extraction from conversations. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 462–467.

Marie W Meteer, Ann A Taylor, Robert MacIntyre, and Rukmini Iyer. 1995. *Dysfluency annotation stylebook for the switchboard corpus*. University of Pennsylvania Philadelphia, PA.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Dmitrijs Milajevs and Matthew Purver. 2014. Investigating the contribution of distributional semantic information for dialogue act classification. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)*, pages 40–47, Gothenburg, Sweden, April. Association for Computational Linguistics.

Jeremy Morris and Eric Fosler-Lussier. 2006. Combining phonetic attributes using conditional random fields. In *Ninth International Conference on Spoken Language Processing*.

Daniel Ortega and Ngoc Thang Vu. 2017. Neural-based context representation learning for dialog act classification. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 247–252, Saarbrücken, Germany, August. Association for Computational Linguistics.

Volha Petukhova and Harry Bunt. 2009. Who's next? speaker-selection mechanisms in multiparty dialogue. In *Workshop on the Semantics and Pragmatics of Dialogue*.

Vipul Raheja and Joel Tetreault. 2019. Dialogue Act Classification with Context-Aware Self-Attention. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3727–3733, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May. ELRA.

Nils Reimers and Iryna Gurevych. 2017. Optimal hyperparameters for deep lstm-networks for sequence labeling tasks. *arXiv preprint arXiv:1707.06799*.

Klaus Ries. 1999. Hmm and neural network based speech act detection. In *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No. 99CH36258)*, volume 1, pages 497–500. IEEE.

Harvey Sacks, Emanuel A. Schegloff, and Gail Jefferson. 1974. A simplest systematics for the organization of turn-taking for conversation. *Language*, 50(4):696–735.

John Rogers Searle. 1969. *Speech acts: An essay in the philosophy of language*, volume 626. Cambridge university press.

Guokan Shang, Wensi Ding, Zekun Zhang, Antoine Tixier, Polykarpos Meladianos, Michalis Vazirgiannis, and Jean-Pierre Lorré. 2018. Unsupervised abstractive meeting summarization with multi-sentence compression and budgeted submodular maximization. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 664–674, Melbourne, Australia, July. Association for Computational Linguistics.

Guokan Shang, Antoine Jean-Pierre Tixier, Michalis Vazirgiannis, and Jean-Pierre Lorré. 2019. Energy-based self-attentive learning of abstractive communities for spoken language understanding. *arXiv preprint arXiv:1904.09491*.

Sheng-syun Shen and Hung-yi Lee. 2016. Neural attention models for sequence classification: Analysis and application to key term extraction and dialogue act detection. *arXiv preprint arXiv:1604.00077*.

Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26(3):339–374.

Dinoj Surendran and Gina-Anne Levow. 2006. Dialog act tagging with support vector machines and hidden markov models. In *Ninth International Conference on Spoken Language Processing*.

Charles Sutton, Andrew McCallum, et al. 2012. An introduction to conditional random fields. *Foundations and Trends® in Machine Learning*, 4(4):267–373.

Maryam Tavafi, Yashar Mehdad, Shafiq Joty, Giuseppe Carenini, and Raymond Ng. 2013. Dialogue act recognition in synchronous and asynchronous conversations. In *Proceedings of the SIGDIAL 2013 Conference*, pages 117–121, Metz, France, August. Association for Computational Linguistics.

Quan Hung Tran, Ingrid Zukerman, and Gholamreza Haffari. 2017. A hierarchical neural model for learning sequences of dialogue acts. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 428–437, Valencia, Spain, April. Association for Computational Linguistics.

Anand Venkataraman, Yang Liu, Elizabeth Shriberg, and Andreas Stolcke. 2005. Does active learning help automatic dialog act tagging in meeting data? In *Ninth European Conference on Speech Communication and Technology*.

Andrew Viterbi. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE transactions on Information Theory*, 13(2):260–269.

Nick Webb, Mark Hepple, and Yorick Wilks. 2005. Dialogue act classification based on intra-utterance features. In *Proceedings of the AAAI Workshop on Spoken Language Understanding*, volume 4, page 5. Citeseer.

Genta Indra Winata, Zhaojiang Lin, Jamin Shin, Zihan Liu, and Pascale Fung. 2019. Hierarchical meta-embeddings for code-switching named entity recognition. *arXiv preprint arXiv:1909.08504*.

Hang Yan, Bocao Deng, Xiaonan Li, and Xipeng Qiu. 2019. Tener: Adapting transformer encoder for name entity recognition. *arXiv preprint arXiv:1911.04474*.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, San Diego, California, June. Association for Computational Linguistics.

Jie Yang, Shuailong Liang, and Yue Zhang. 2018. Design challenges and misconceptions in neural sequence labeling. *arXiv preprint arXiv:1806.04470*.

Linhao Zhang and Houfeng Wang. 2019. Using bidirectional transformer-crf for spoken language understanding. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 130–141. Springer.

Matthias Zimmermann. 2009. Joint segmentation and classification of dialog acts using conditional random fields. In *Tenth Annual Conference of the International Speech Communication Association*.

Appendix A. Worst and best case analysis

In this section, we interpret the confusion matrices for the 10 DA labels that our model best predicted (Fig. 5) and worst predicted (Fig. 6), in comparison with the base model, always on the right. Inspecting the matrices reveals that our model is most useful for the DAs requiring speaker-change awareness, which confirms the effectiveness of our modification of the CRF layer. It also shows that our model brings improvement where it is most necessary, i.e., for the most difficult and rare DAs.

Relative differences. For the 10 DA labels best predicted by our model, the average performance gain compared to the base model is equal to **5.38** (shown in Table 4), whereas the drop in performance for the 10 DAs worst predicted by our model is lower, only equal to **4.87**. Thus, when it improves performance, our model does so with a greater margin than when it decreases performance. This fact is hidden when simply looking at the global accuracy over the 42 DA labels, because the 10 best DAs for our model only correspond to 20.2% of all annotations, whereas the 10 worst account for almost 40% of all annotations.

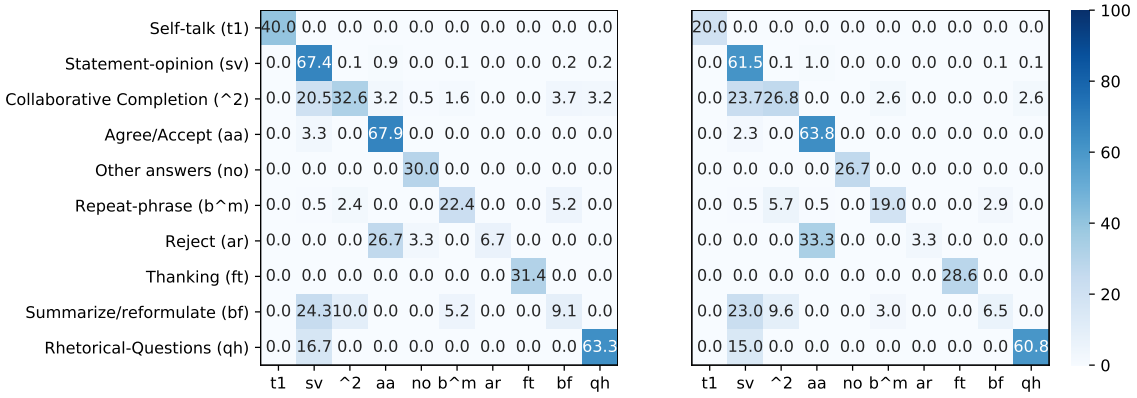

Figure 5: Normalized confusion matrices, averaged over 10 runs, for the 10 DA labels **best** predicted by our model (20.2% of all annotations). Left: our model, right: base model.

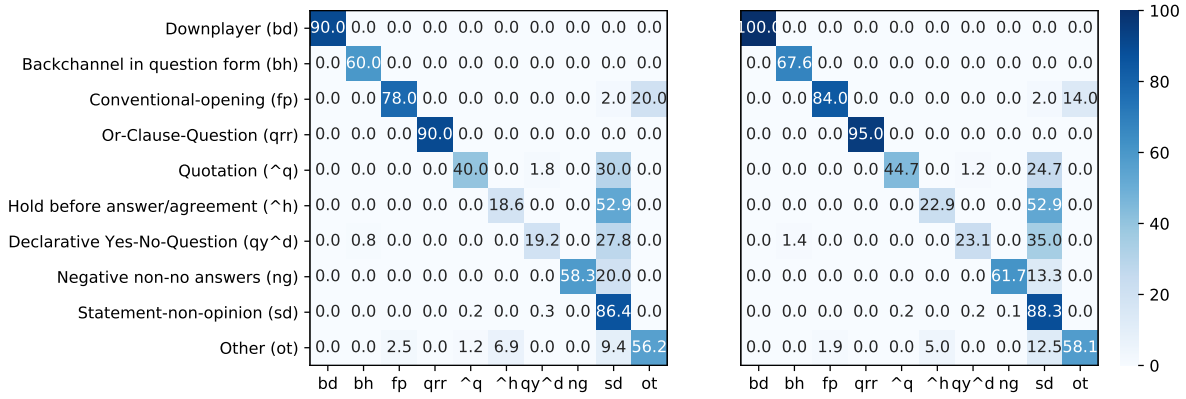

Figure 6: Normalized confusion matrices, averaged over 10 runs, for the 10 DA labels **worst** predicted by our model (39.6% of all annotations). Left: our model, right: base model.

Absolute differences. It is also interesting to note that the 10 DAs that our model best predicted are all very difficult DAs, for which the performance of the base model is very low in the first place: **31.7**, on average. These DAs are also rare: they only correspond to 20.2% of all annotations. Our model raises the average accuracy on these labels to **37.08**. On the other hand, the 10 DAs that are worst predicted by our model are more frequent DAs (40% of all annotations), for which the performance of the base model is already quite high: **64.54**, on average. And although our model is not as good as the base model on these DAs, it still reaches a decent average performance of **59.67**. Therefore, our model provides a performance boost where it is most necessary (difficult and rare DAs), and wherever it fails, it still provides decent accuracy levels.

10 best DAs for our model. Our model outperforms the base model by a very large margin of 20.0% (20.0%→40.0%) for Self-talk (`t1`, the speaker talks to him/herself). It makes a lot of sense, as the accurate prediction of this DA obviously requires being aware of speaker-change. Similar conclusions can be also drawn for Collaborative Completion (`^2`, one speaker completes the other speaker's utterance), Repeat-phrase (`b^m`, repeating parts of what the previous speaker said), Thanking (`ft`), Summarize/reformulate (`bf`, proposing a summarization or paraphrase of another speaker's talk/point), and Rhetorical-Questions (`qh`, questions asked to make a statement or asked to produce an effect with no answer expected).

10 worst DAs for our model. On the other hand, speaker information does not seem to be crucial to predict the 10 DA labels most often missed by our model. For instance, Conventional-openings (fp) are

always found among the first utterances in a conversation, so there is only a small need for speaker-change awareness in that case. E.g., in this situation with three utterances: (1) "`A: Hi, Wanet (fp)`", (2) "`A: How are you? (fp)`", and (3) "`B: I'm doing fine. (fp)`", utterances 2 and 3 are labeled with `fp`, regardless of speaker-change. Likewise, the need for speaker-change awareness seems very little for the Quotation (^q) and Other (ot) DAs. In other words, among the DAs worst predicted by our model are DAs for which speaker information is not necessary to make an accurate prediction. This makes sense, since the goal of our modified CRF layer is precisely to capture speaker information.

Appendix B. DA label statistics

Dialogue Act (label)	count	frequency	Dialogue Act (label)	count	frequency
Statement-non-opinion (sd)	73873	36.85%	Collaborative Completion (^2)	709	0.35%
Acknowledge/Backchannel (b)	37727	18.82%	Repeat-phrase (b^m)	677	0.34%
Statement-opinion (sv)	25810	12.88%	Open-Question (qo)	647	0.32%
Abandoned/Uninterpretable (%)	15294	7.63%	Rhetorical-Questions (qh)	566	0.28%
Agree/Accept (aa)	10987	5.48%	Hold before answer/agreement (^h)	546	0.27%
Appreciation (ba)	4702	2.35%	Reject (ar)	341	0.17%
Yes-No-Question (qy)	4679	2.33%	Negative non-no answers (ng)	296	0.15%
Non-verbal (x)	3565	1.78%	Signal-non-understanding (br)	295	0.15%
Yes answers (ny)	2995	1.49%	Other answers (no)	284	0.14%
Conventional-closing (fc)	2562	1.28%	Conventional-opening (fp)	225	0.11%
Wh-Question (qw)	1954	0.97%	Or-Clause Question (qrr)	208	0.10%
No answers (nn)	1363	0.68%	Dispreferred answers (arp_nd)	207	0.10%
Response Acknowledgement (bk)	1299	0.65%	3rd-party-talk (t3)	115	0.06%
Hedge (h)	1204	0.60%	Offers, Options, Commits (oo_co_cc)	109	0.05%
Declarative Yes-No-Question (qy^d)	1203	0.60%	Maybe/Accept-part (aap_am)	105	0.05%
Backchannel in question form (bh)	1036	0.52%	Self-talk (t1)	103	0.05%
Quotation (^q)	948	0.47%	Downplayer (bd)	101	0.05%
Summarize/reformulate (bf)	928	0.46%	Tag-Question (^g)	92	0.05%
Other (ot)	876	0.44%	Declarative Wh-Question (qw^d)	80	0.04%
Affirmative non-yes answers (na)	841	0.42%	Apology (fa)	78	0.04%
Action-directive (ad)	740	0.37%	Thanking (ft)	74	0.04%

Table 3: Counts and frequencies of the 42 DA labels in the SwDA dataset. There are 200444 utterances in total.