Multi-Task Self-Supervised Learning for Disfluency Detection

Shaolei Wang, Wanxiang Che, Qi Liu, Pengda Qin, Ting Liu, William Yang Wang

School of Computer Science and Technology
Harbin Institute of Technology, Harbin, China

Research Center for Social Computing and Information Retrieval
理解语言 认知社会

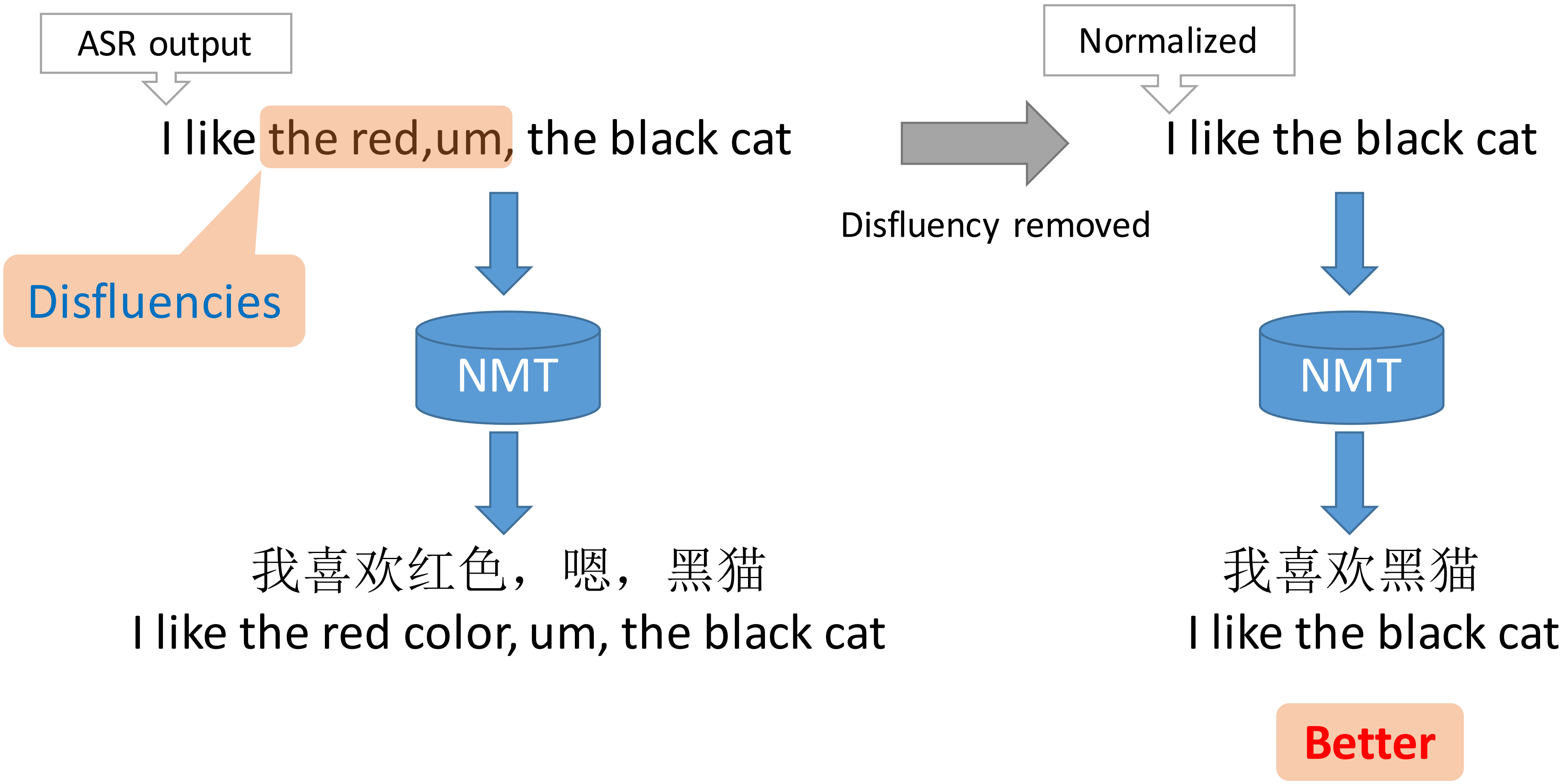
Disfluency Detection

- The transcribed speech text is mostly disfluent

$$\text{I want a flight [} \underbrace{\textit{to Boston}}_{\text{RM}} + \underbrace{\{\textit{um}\}}_{\text{IM}} \underbrace{\text{to Denver}}_{\text{RP}} \text{]}$$

Figure 1: A sentence from the English Switchboard corpus with disfluencies annotated. RM=Reparandum, IM=Interregnum, RP=Repair. The preceding RM is corrected by the following RP.

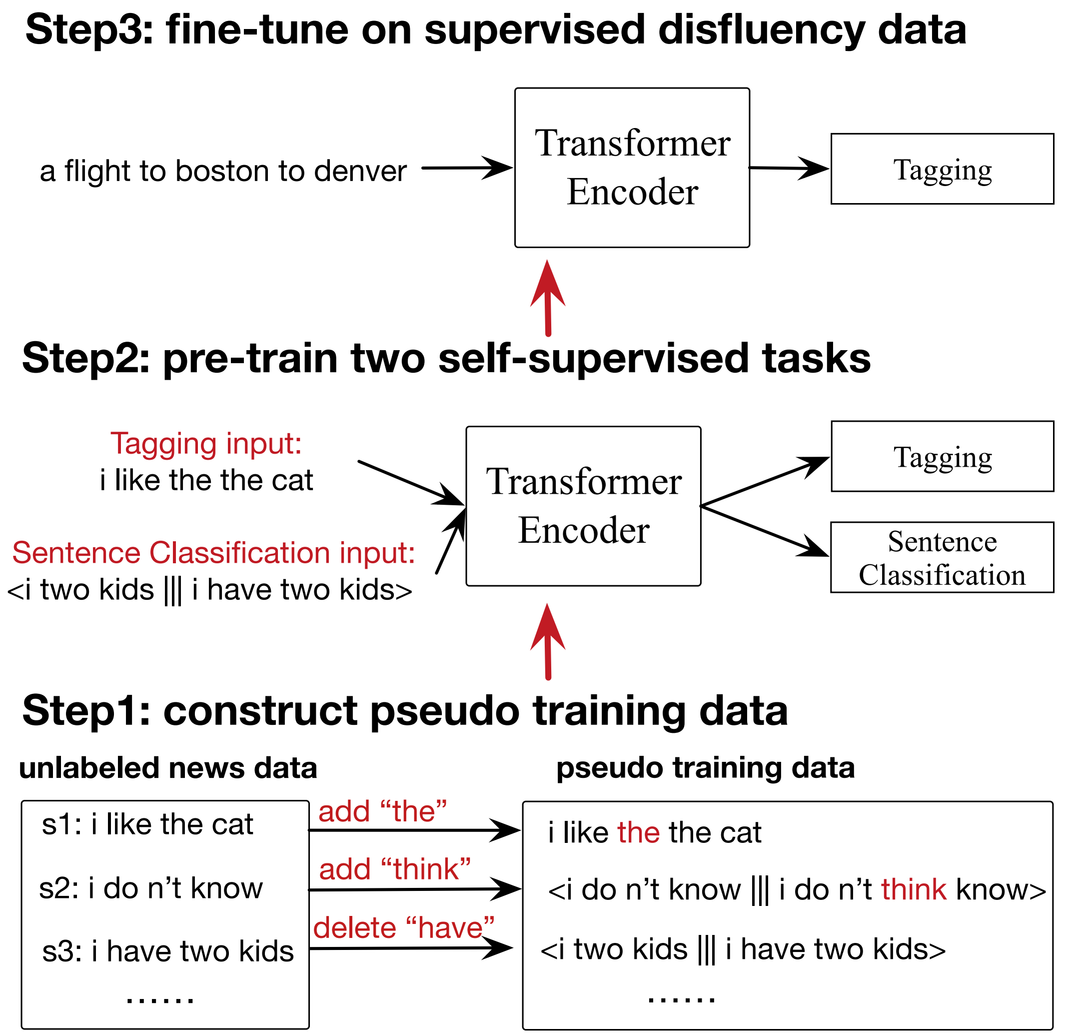
Disfluency Effect on Machine Translation



Our Motivations

- tackle the training data bottleneck
  - construct large-scale pseudo training data by randomly adding or deleting words from unlabeled news data
  - propose two self-supervised pre-training task

Our Model


Step3: fine-tune on supervised disfluency data
a flight to boston to denver
Transformer Encoder
Tagging
Step2: pre-train two self-supervised tasks
Tagging input:
i like the the cat
Sentence Classification input:
<i two kids ||| i have two kids>
Transformer Encoder
Tagging
Sentence Classification
Step1: construct pseudo training data
unlabeled news data
s1: i like the cat
s2: i do n't know
s3: i have two kids
......
add "the"
add "think"
delete "have"
pseudo training data
i like the the cat
<i do n't know ||| i do n't think know>
<i two kids ||| i have two kids>
......

Our Model

- Construct pseudo training data
  - Type1: S_{disf}
    - *Repetition*(*k*):the *m* (randomly selected from *one* to *six*) words starting from the position *k* are repeated.
    - *Inserting*(*k*) : randomly pick a *m*-gram (*m* is randomly selected from *one* to *six*) from the news corpus and insert it to the position *k*.
    - Eg: I like the cat → I think like the the cat
  - Type2: S_{del}
    - *Delete*(*k*) : for selected position *k*, *m* (randomly selected from *one* to *six*) words starting from this position are deleted.
    - Eg: he has two kids → he two kids

Our Model

- Tagging Task
  - detect the added noisy words in S_{disf}

eg: input: I think like the the cat

output: O D O D O O

- Classification Task
  - distinguish original sentences from grammatically-incorrect sentences.

eg: inout: <he has two kids ||| he two kids>

output: del_1

Our Model

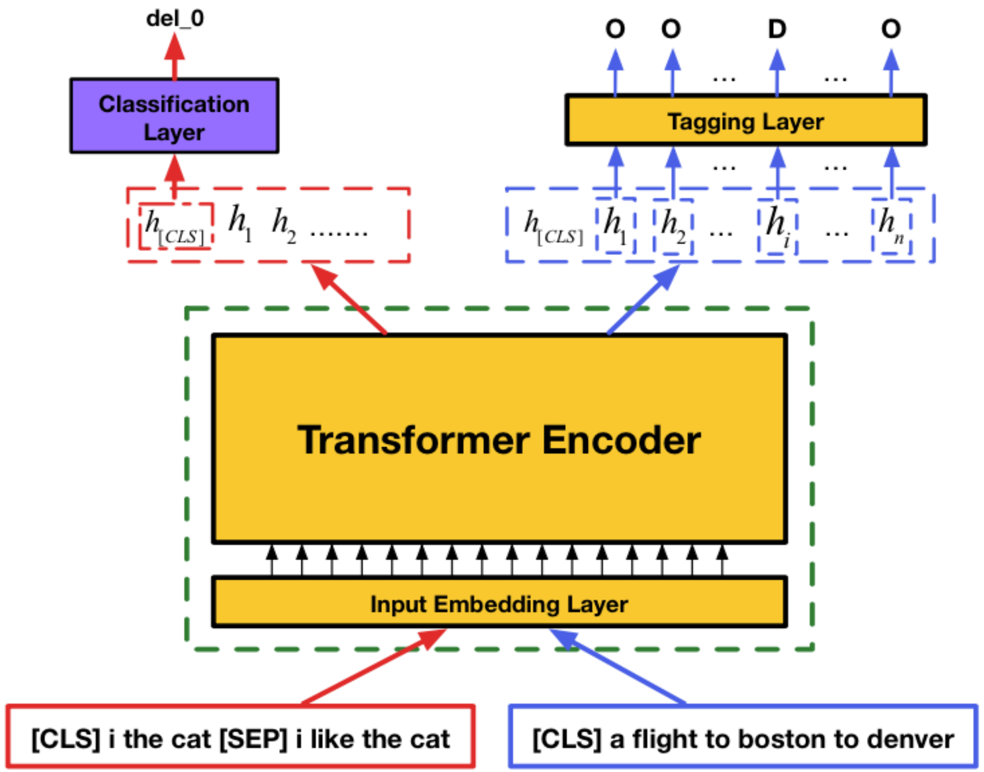


Figure 3: Model structure. The parameters of input embedding layer I, encoder layer E, and tagging layer T (yellow box) are shared among pre-training and fine-tuning

Experimental Setting

- Dataset
  - Pre-training data: 12 million
    - 3 million for tagging task
    - 9 million for classification task
  - English Switchboard corpus
    - About 100000 sentences for training data
- Model Size
  - 512 hidden units, 8 heads, 6 hidden layers

Experiment results

- Experiment results on the development and test data of English Switchboard data

Method	Full						1000 sents					
	Dev			Test			Dev			Test		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Transition-based	92.2	84.7	88.3	92.1	84.1	87.9	82.2	57.4	67.6	81.2	56.7	66.8
Transformer-based	86.5	70.4	77.6	86.1	71.5	78.1	78.2	51.3	62.0	79.1	51.1	62.1
Our self-supervised	92.9	88.1	**90.4**	93.4	87.3	**90.2**	90.0	82.8	**86.3**	88.6	83.7	**86.1**

Experiment results

- Comparison with the previous state-of-the-art methods

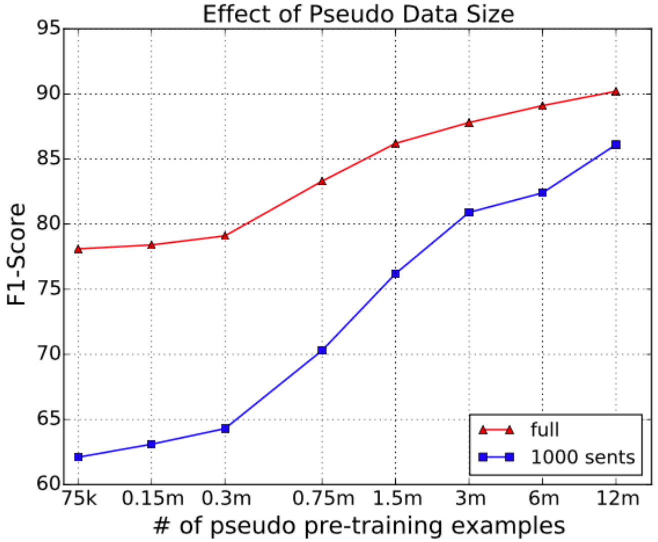
Method	P	R	F1
UBT (Wu et al. 2015)	90.3	80.5	85.1
Semi-CRF (Ferguson et al., 2015)	90.0	81.2	85.4
Bi-LSTM (Zayats et al., 2016)	91.8	80.6	85.9
LSTM-NCM (Lou and Johnson 2017)	-	-	86.8
Transition-based (Wang et al. 2017)	91.1	84.1	87.5
Our self-supervised (1000 sents)	88.6	83.7	86.1
Our self-supervised (Full)	**93.4**	**87.3**	**90.2**

Experiment results

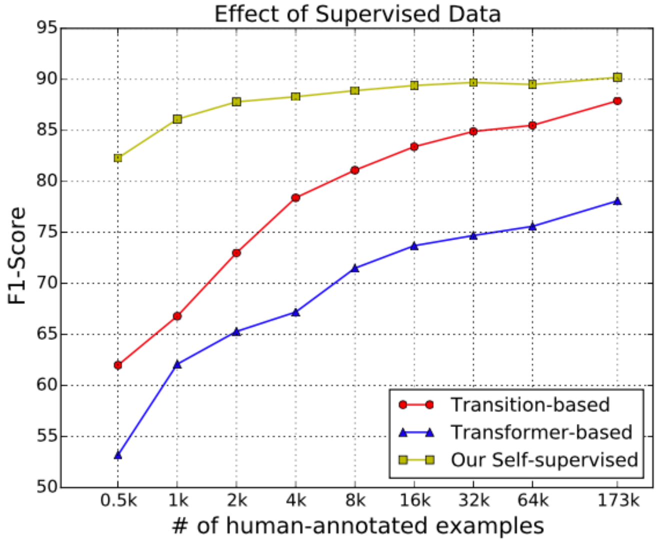
- Ablation over the two self-supervised tasks

Method	Full			1000 sents		
	P	R	F1	P	R	F1
Random-Initial	86.1	71.5	78.1	79.1	51.1	62.1
Tagging	91.8	84.0	87.7	85.1	79.6	82.3
Classification	91.2	83.1	87.0	83.2	78.3	80.7
Multi-Task	**93.4**	**87.3**	**90.2**	**88.6**	**83.7**	**86.1**

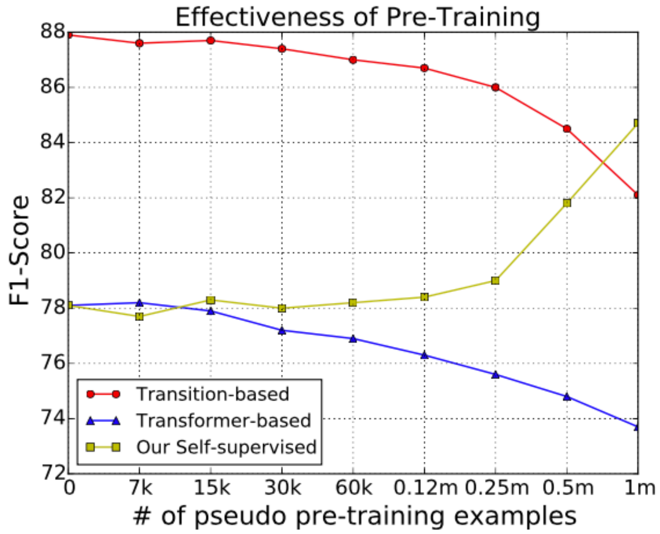
Experiment results



(a)



(b)



(c)

Experiment results

Comparison with BERT

Method	F1 (Full)	F1 (1000 sents)
Random-Initial	78.1	62.1
BERT-fine-tune	90.1	82.4
Our self-supervised	90.2	86.1
Combine	**91.4**	**87.8**

Table 7: Comparison with BERT. "random-initial" means training transformer network on gold disfluency detection data with random initialization. "combine" means concatenating hidden representations of BERT and our self-supervised models for fine-tuning.

Conclusion

- Propose two self-supervised tasks for disfluency detection to tackle the training data bottleneck.

- Experimental results show that our approach can achieve competitive performance compared to the previous systems by using less than 1% (1000 sentences) of the training data

Thank you!