Disfluency Correction using Unsupervised and Semi-supervised Learning

Nikhil Saini[*1], **Drumil Trivedi**[*1], **Shreya Khare**[2], **Tejas I. Dhamecha**[2], **Preethi Jyothi**[1],
Samarth Bharadwaj[2] **and Pushpak Bhattacharyya**[1]

[1]Indian Institute of Technology Bombay
{nikhilra,drumilt,pjyothi,pb}@cse.iitb.ac.in
[2]IBM Research - India
{skhare34,tidhamecha,samarth.b}@in.ibm.com

Abstract

Spoken language is different from the written language in its style and structure. Disfluencies that appear in transcriptions from speech recognition systems generally hamper the performance of downstream NLP tasks. Thus, a disfluency correction system that converts disfluent to fluent text is of great value. This paper introduces a disfluency correction model that translates disfluent to fluent text by drawing inspiration from recent encoder-decoder unsupervised style-transfer models for text. We also show considerable benefits in performance when utilizing a small sample of 500 parallel disfluent-fluent sentences in a semi-supervised way. Our unsupervised approach achieves a BLEU score of 79.39 on the Switchboard corpus test set, with further improvement to a BLEU score of 85.28 with semi-supervision. Both are comparable to two competitive fully-supervised models.

1 Introduction

Disfluencies are disruptions to the regular flow of speech, typically occurring in conversational speech. They include filler pauses such as *uh* and *um*, word repetitions, irregular elongations, discourse markers, conjunctions, and restarts. For example, the disfluent sentence "well we're actually uh we're getting ready" has its fluent form as, "we're getting ready". Here, the words highlighted in green, blue and red refer to discourse, filler and restart disfluencies, respectively.

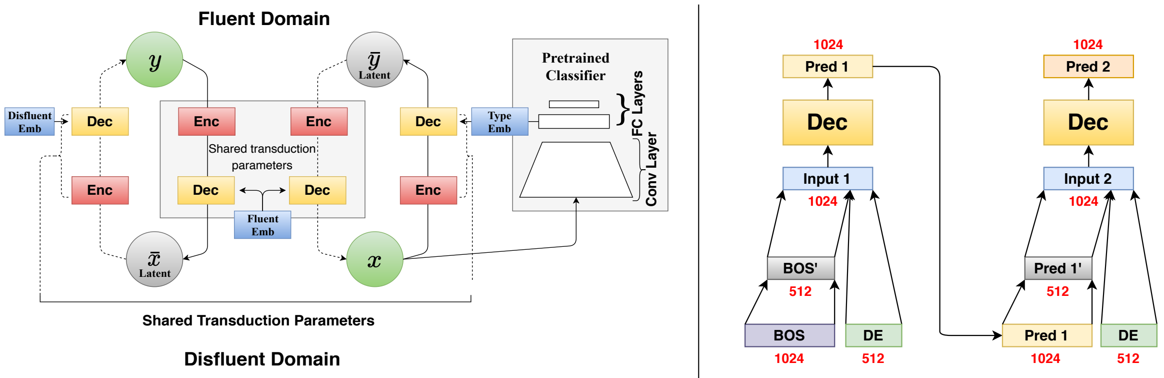
Disfluencies in the text can alter its syntactic and semantic structure, thereby adversely affecting the performance of downstream NLP tasks such as information extraction, summarization, translation, and parsing (Charniak and Johnson, 2001; Johnson and Charniak, 2004). These tasks also employ pretrained language models that are typically trained to expect fluent text. This motivates the need for disfluency correction systems that convert disfluent to fluent text. Prior work has predominantly focused on the problem of disfluency detection (Zayats et al., 2016; Wang et al., 2018; Dong et al., 2019). Inspired by recent work on unsupervised machine translation and style-transfer models for text, we propose an unsupervised encoder-decoder based model to tackle the problem of disfluency correction. Our model does not require access to a parallel corpus of disfluent and fluent sentences. We also show a semi-supervised variant of our model that uses a small amount of parallel disfluent-fluent text and significantly improves performance. To our knowledge, this is the first work to use state-of-the-art unsupervised models for the task of disfluency correction. Our main contributions are as follows:

- We cast the problem of disfluency correction as one of translation from disfluent to fluent text and we propose an unsupervised transformer-based encoder-decoder model for disfluency correction.

- We compare and contrast an unsupervised and semi-supervised approach for disfluency correction, where the latter has access to a small amount of parallel text. We also implement fully-supervised methods as a skyline and show how our models come very close in performance to these approaches, which are very resource-intensive and require large amounts of parallel text.

- We show detailed ablation analyses across disfluency types and present a qualitative study of disfluency corrections that our model can achieve.

[*] Joint first authors



(a) Style-transfer

(b) Use of domain embedding

Figure 1: Illustration of (a) Style transfer model modified to use type embedding drawn from a pretrained CNN classifier. (b) Conditioning on domain embeddings in the transformers' decoder. Pred(i) and Input(i) are the decoder's prediction and input to the decoder at the i_{th} time-step respectively.

2 Related work

Current literature has primarily focused on disfluency detection in both speech and text in fully supervised settings (Wang et al., 2016; Georgila et al., 2010; Zayats et al., 2014; Tran et al., 2019; Wang et al., 2018; Bach and Huang, 2019; Zayats et al., 2016; Lou and Johnson, 2020a). The grammatical error correction (Omelianchuk et al., 2020) approach does not perform well on the disfluency correction tasks. In most cases, simply removing disfluencies from an utterance can render the sentence ill-formed. More meaningful and syntactically well-formed utterances are generated by performing automatic disfluency removal from speech (Kaushik et al., 2010; Lou and Johnson, 2020b) and text (Wang et al., 2010; Honal and Schultz, 2005; Hassan et al., 2014). With the popularity of end-to-end spoken translation systems, several works translate fluent utterances from disfluent speech (Salesky et al., 2018; Ansari et al., 2020; Fukuda et al., 2020) or disfluent text (Cho et al., 2013; Saini et al., 2020; Cho et al., 2016). Most of these approaches work in a supervised setting or mitigate the lack of parallel disfluent-fluent text via data augmentation, model design, incorporating domain knowledge of the language, or using multi-lingual NMT. (Salesky et al., 2019) proposes a system for conversational speech translation with the joint removal of disfluencies.

3 Our Approach

We draw inspiration from unsupervised neural machine translation models (Lample et al., 2017) and style transfer models (He et al., 2020) to design the disfluency correction model illustrated in Figure 1a. It consists of a single encoder and a single decoder, used to translate in both directions, i.e., from disfluent to fluent text and vice-versa. The decoder is additionally conditioned using a *domain embedding* to convey the direction of translation, signifying whether the input to the encoder is a fluent or disfluent sentence. More details about our framework are described below.

3.1 Unsupervised Disfluency Correction

Figure 1a shows the two directions of translation. The model obtains latent disfluent and latent fluent utterances from the non-parallel fluent and disfluent sentences, respectively, which are further reconstructed back into fluent and disfluent sentences. We employ a backtranslation-based objective, followed by reconstruction for both domains, i.e., disfluent and fluent text. For every mini-batch of training, soft translations for a domain are first generated (denoted by $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$ in Figure 1a), and are subsequently translated back into their original domains to reconstruct the mini-batch of input sentences. The sum of token-level cross-entropy losses between the input and the reconstructed output serves as the reconstruction loss.

Borrowing from prior work on unsupervised style transfer model (He et al., 2020), the decoder is conditioned on a domain embedding that specifies the direction of translation. In this work, we employ two types of embeddings: A vanilla *binary domain embedding* that takes a bit as input to indicate whether the input text is fluent or disfluent and a *classifier-based domain embedding*. The latter is obtained from a trained standalone CNN-based classifier (Kim, 2014) that predicts the disfluency type of a disfluent input sentence. (Here, we assume that disfluency type labels are available for the disfluent sentences in our training data.) The classifier's penultimate layer acts as our classifier embedding, which is further used to condition the decoder. We hypothesize that additional information about disfluency types via the classifier-based embedding might help guide the process of disfluency correction better.

Furthermore, similar to the noise models adopted by (He et al., 2020; Lample et al., 2017), a randomly sampled noisy version of every sentence in the input mini-batch is fed to the model, forcing it to behave like a denoising auto-encoder. We use noise perturbations (Lample et al., 2017) in the form of word-shuffle(α), word-blank(β) and word-dropout(γ) operations.

We explore two choices to implement our encoder-decoder modules: 1) BiLSTM-based (Bahdanau et al., 2015) and 2) Transformer-based (Vaswani et al., 2017). For the BiLSTM model, as proposed by (He et al., 2020), the BOS vector, i.e., the input to the decoder at the first time-step, is replaced by the domain embeddings. In the Transformer model, this conditioning needs to be carefully done. Figure 1b illustrates how we conditioned the transformer-based decoder. Dimensionality reduced word embedding is concatenated with the domain embedding *DE* at every time-step(*t*) to form the input for the decoder.

3.2 Semi-Supervised Disfluency Correction

Our unsupervised disfluency correction model can be easily fine-tuned using small amounts of parallel text, when available, lending itself to semi-supervised learning. The encoder-decoder modules are initialized using the unsupervised training described in the previous section and further fine-tuned with a supervised cross-entropy loss using small amounts of parallel disfluent-fluent text. We do not use domain embeddings during semi-supervised training; the inference is done as in the unsupervised model, i.e., with domain embeddings.

4 Experiments and Results

In this work, we use the Switchboard corpus (Godfrey et al., 1992) that includes telephonic conversations and their disfluency annotations (Schriberg, 1994; Zayats et al., 2014). We create a 70:15:15 train, test, and validation split. The train set contains 110,964 sentences, whereas validation and test sets have 11,889 disfluent-fluent sentence pairs.

Model	BLEU		METEOR	
	Dev	Test	Dev	Test
Disfluent	70.98	71.53	57.02	57.19
US (BiLSTM)	61.26	62.64	48.31	49.13
US (Transformer)	78.72	79.39	56.59	57.25
SS (Transformer)	83.85	85.28	57.77	58.35
Seq2Seq	87.23	88.08	56.65	59.36
BART	89.27	90.08	62.17	63.01

Table 1: BLEU and METEOR scores on the Switchboard dev and test sets. US and SS represent our unsupervised and semi-supervised approaches, respectively.

4.1 Implementation Details

Our BiLSTM model uses a single layer of recurrent units of hidden size 750 with max-pooling over a window size of 5. The noise perturbation parameters, α, β, γ were tuned on the validation set and set to 0. The model was trained for 15 epochs with 10 for annealing, using mini-batches of size 32, with Adam optimizer (Kingma and Ba, 2015) and a learning rate 0.01 linearly scheduled with the rate of decrements of 0.5. Empirically, we also found it essential to allow gradients to pass through the backtranslations to generate meaningful sentences.

The transformer model uses 8 attention heads, word embedding and domain embedding dimensionalities of 1024 and 512. The noise perturbation parameters, α, β, γ are set as 3, 0.2, 0.1. Adam optimizer is used with an initial learning rate of 0.00001, with a linear scheduler and 10 warm-up steps. We used mini-batches of size 32. Dropout (Gal and Ghahramani, 2016) and label-smoothing (Szegedy et al., 2016) values were 0.3 and 0.1, respectively.

4.2 Results

Table 1 shows BLEU and METEOR scores between the gold fluent and the disfluency corrected output from five different models. We train two fully supervised skylines, based on Seq2Seq (Sutskever et al., 2014) and BART (Lewis et al., 2019), to compare against our approaches. The BLEU score using original disfluent text as the hypothesis is 71.53. The two supervised skylines use 55K pairs of parallel disfluent-fluent sentences during training and yield up to 90 BLEU score. In comparison, the unsupervised approach yields up to 80 BLEU scores without any parallel data. Fine-tuning the unsupervised model with a small parallel corpus containing only 554 pairs (i.e. two orders

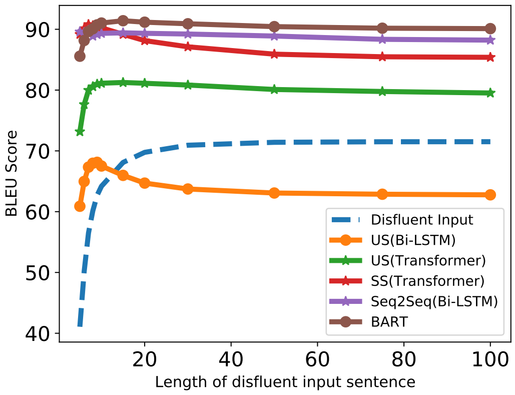


Figure 2: BLEU scores vs. Input lengths.

of magnitude smaller than the complete set of 55K pairs) significantly bridges this gap and yields up to 85 BLEU score. In terms of METEOR, the score using original disfluent text as the hypothesis is 57.19. The difference between unsupervised and supervised approaches is much smaller, indicating that with respect to the adequacy or content preservation, these approaches perform at par. These results also show that the last few additional BLEU points (i.e., the difference between BART and SS) come at a high cost with having to create a large parallel corpus.

We obtain 77.34 and 77.97 BLEU on the dev and test sets using binary embeddings, respectively, whereas the disfluency-type classifier embedding yields 78.72 and 76.90 on the dev and test sets. The classifier embeddings do marginally improve performance. However, the BLEU scores obtained using the binary embeddings are almost comparable, which shows that our proposed model can effectively use even non-parallel text without any disfluency type labels.

Sentence Length: Figure 2 shows BLEU scores as a function of maximum sentence length on the test set. The BLEU score is highest for the utterances smaller than ten tokens; on longer sentences, the BLEU scores drop. This trend is uniform across all models. Our transformer-based model significantly outperforms the BiLSTM-based model on utterances of all lengths. Interestingly, our semi-supervised approach is very similar in performance to the fully supervised approach for smaller (<10 token) utterances.

Semi-supervised Learning: Table 2 shows the performance when our unsupervised model is fine-tuned with varying amounts of parallel text.

#Sentences	Percentage(%)	Dev	Test
0	(Unsupervised) 0	78.72	79.39
554	1	83.85	85.28
2774	5	84.67	86.03
5548	10	84.98	86.12
13870	25	85.88	87.04
27741	50	86.10	87.90
55482	100	87.16	88.22

Table 2: Effect of fine-tuning with a varying amount of supervised parallel corpus to fine-tune our model trained in unsupervised manner; in effect, results of semi-supervised training.

	Set	US BiLSTM	US Trans.	SS Trans.	Seq2Seq	BART
all	Dev	61.26	78.72	84.10	87.23	89.27
	Test	62.64	79.39	85.28	88.08	90.08
Conj	Dev	62.68	80.17	84.65	87.63	88.98
	Test	63.60	80.18	86.24	89.13	89.79
filler	Dev	56.96	76.86	81.01	85.01	87.41
	Test	58.45	77.47	82.16	85.92	88.59
restart	Dev	53.76	72.39	78.13	82.24	84.99
	Test	54.92	73.06	79.52	82.84	85.60
disc	Dev	53.84	71.91	81.05	84.64	87.52
	Test	55.39	73.19	82.30	85.93	88.57
edit	Dev	49.06	63.20	78.28	82.86	85.45
	Test	52.51	64.35	80.60	85.82	87.13
aside	Dev	37.07	48.56	41.25	53.98	53.88
	Test	37.25	45.71	51.65	56.37	55.68

Table 3: Disfluency type specific BLEU scores. (Trans.: Transformer, conj: conjunctions and disc: discourse disfluencies).

By having access to only 554 parallel pairs (i.e., 1% total pairs), the performance improves by an impressive 5.89 BLEU on the test set. While BLEU improvements are a monotonically increasing function of the amount of parallel text, we see a trend of diminishing returns soon after the 1% mark.

Performance Across Disfluency Types: Intuitively, certain types of disfluencies (e.g., fillers) are easier to correct than others (e.g., edits). Table 3 reports the BLEU scores from all our models across disfluency types. Conjunctions and discourse disfluencies mark the easy end of the correction spectrum, while edits and asides mark the problematic end. (Edits are also hard to correct because of the lack of training data.)

Qualitative Analysis: Table 4 shows examples using five different models along with corresponding disfluent and fluent sentences. All five models can remove simpler disfluencies (e.g., fillers and

	Disfluent	BART	Seq-to-Seq	US(Bi-LSTM)	US(Trans.)	SS	Fluent
disc., filler	so uh been a different turn	been a different turn	been a different turn	been a different turn	been a different turn	been a different turn	been a different turn
conj., rep.	but i i i find this whole	i find this whole	i find this whole	anyway i find it all	i find this whole	i find this whole	i find this whole
restart	it's you're you're taking words and developing a picture in your mind	you're taking words and developing a picture in your mind	you're taking words and developing a picture in your mind	it's you're taking chicken and tobacco words in a mind	it's taking words and developing and a picture in your mind	it's taking words and developing a picture in your mind	you're taking words and developing a picture in your mind
conj., disc., restart	and then you you know you had i think you had to pick it by by by the end of the second you had to pick some sort of major	then you think you had to pick it by the end of the second you had to pick some sort of major	you had to pick it by by the end of the second you had to pick some sort of major	then you think you had i think you had to pick it by by the end of the second you had to pick some sort of major	then you had to pick some sort of major	you had to pick of it by the end of you had to pick some major	by the end of the second you had to pick some sort of major
aside	i forgot sally's last name anyway it's a couple	i forgot sally's last name anyway it's a couple	i forgot seen last name anyway it's a couple	gosh i forgot last name it's a couple of years	i forgot wordstart last name anyway it's a couple	i forgot harry name anyway it's a couple	it's a couple

Table 4: Analysis of generated text across all models. (disc.: Discourse; conj.: Conjunction; rep.:repetition US: Unsupervised; SS: Semi-supervised; Trans.: Transformer.)

discourse) in shorter sentences. Conjunctions and repetitions are removed by all models except the unsupervised BiLSTM model. The third example shows how the transformer model is much better than the BiLSTM model in terms of content retention and adequacy. It also highlights the better fluency of the semi-supervised model than the unsupervised model. The fourth example illustrates the increased complexity due to the presence of multiple disfluency types(*conjunction, discourse, restart*) in a single utterance. The fifth example illustrates a case of an aside, which is difficult for all models. It shows how even the supervised models fail to detect the disfluent phrase *"i forgot sally's last name anyway"*.

5 Conclusion

We propose an unsupervised disfluency correction model drawing motivation from prior work on unsupervised machine translation and style transfer. We investigate two kinds of domain embeddings for our model. We also present a semi-supervised disfluency correction approach. We finetune our model using only about 500 parallel sentences, which comes very close in performance (based on BLEU scores) to a state-of-the-art, fully supervised system.

References

Ebrahim Ansari, Amittai Axelrod, Nguyen Bach, Ondřej Bojar, Roldano Cattoni, Fahim Dalvi, Nadir Durrani, Marcello Federico, Christian Federmann, Jiatao Gu, Fei Huang, Kevin Knight, Xutai Ma, Ajay Nagesh, Matteo Negri, Jan Niehues, Juan Pino, Elizabeth Salesky, Xing Shi, Sebastian Stüker, Marco Turchi, Alexander Waibel, and Changhan Wang. 2020. FINDINGS OF THE IWSLT 2020 EVALUATION CAMPAIGN. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 1–34, Online. Association for Computational Linguistics.

Nguyen Bach and Fei Huang. 2019. Noisy bilstm-based models for disfluency detection. *Proceedings of Interspeech*, pages 4230–4234.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations*.

Eugene Charniak and Mark Johnson. 2001. Edit detection and parsing for transcribed speech. In *Second Meeting of the North American Chapter of the Association for Computational Linguistics*.

Eunah Cho, Thanh-Le Ha, and Alex H. Waibel. 2013. Crf-based disfluency detection using semantic features for german to english spoken language translation.

Eunah Cho, J. Niehues, Thanh-Le Ha, and A. Waibel. 2016. Multilingual disfluency removal using nmt.

Qianqian Dong, Feng Wang, Zhen Yang, Wei Chen, Shuang Xu, and Bo Xu. 2019. Adapting translation models for transcript disfluency detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:6351–6358.

Ryo Fukuda, Katsuhito Sudoh, and Satoshi Nakamura. 2020. NAIST's machine translation systems for IWSLT 2020 conversational speech translation task. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 172–177, Online. Association for Computational Linguistics.

Yarin Gal and Zoubin Ghahramani. 2016. A theoretically grounded application of dropout in recurrent neural networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS, pages 1027–1035, Red Hook, NY, USA. Curran Associates Inc.

Kallirroi Georgila, Ning Wang, and Jonathan Gratch. 2010. Cross-domain speech disfluency detection. In *Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 237–240. Association for Computational Linguistics.

J. J. Godfrey, E. C. Holliman, and J. McDaniel. 1992. Switchboard: telephone speech corpus for research and development. In *[Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 517–520 vol.1.

Hany Hassan, L. Schwartz, Dilek Z. Hakkani-Tür, and G. Tür. 2014. Segmentation and disfluency removal for conversational speech translation. In *INTERSPEECH*.

Junxian He, Xinyi Wang, Graham Neubig, and Taylor Berg-Kirkpatrick. 2020. A probabilistic formulation of unsupervised text style transfer. In *International Conference on Learning Representations*.

Matthias Honal and Tanja Schultz. 2005. Automatic disfluency removal on recognized spontaneous speech-rapid adaptation to speaker-dependent disfluencies. In *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, volume 1, pages I–969. IEEE.

Mark Johnson and Eugene Charniak. 2004. A tag-based noisy channel model of speech repairs. pages 33–39.

Mayank Kaushik, Matthew Trinkle, and Ahmad Hashemi-Sakhtsari. 2010. Automatic detection and removal of disfluencies from spontaneous speech. In *Proceedings of the Australasian International Conference on Speech Science and Technology*, volume 70.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Guillaume Lample, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2017. Unsupervised machine translation using monolingual corpora only. In *International Conference on Learning Representations*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.

Paria Lou and Mark Johnson. 2020a. Improving disfluency detection by self-training a self-attentive model. pages 3754–3763.

Paria Jamshid Lou and Mark Johnson. 2020b. End-to-end speech recognition and disfluency removal.

Kostiantyn Omelianchuk, Vitaliy Atrasevych, Artem Chernodub, and Oleksandr Skurzhanskyi. 2020. GECToR – grammatical error correction: Tag, not rewrite. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 163–170, Seattle, WA, USA â†' Online. Association for Computational Linguistics.

Nikhil Saini, Jyotsana Khatri, Preethi Jyothi, and Pushpak Bhattacharyya. 2020. Generating fluent translations from disfluent text without access to fluent references: IIT Bombay@IWSLT2020. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 178–186, Online. Association for Computational Linguistics.

Elizabeth Salesky, Susanne Burger, Jan Niehues, and Alex Waibel. 2018. Towards fluent translations from disfluent speech. In *IEEE Spoken Language Technology Workshop*, pages 921–926.

Elizabeth Salesky, Matthias Sperber, and Alex Waibel. 2019. Fluent translations from disfluent speech in end-to-end speech translation. *CoRR*, abs/1906.00556.

Elisabeth Schriberg. 1994. Preliminaries to a theory of speech disfluencies.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. Cambridge, MA, USA. MIT Press.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and ZB Wojna. 2016. Rethinking the inception architecture for computer vision.

Trang Tran, Jiahong Yuan, Yang Liu, and Mari Ostendorf. 2019. On the Role of Style in Parsing Speech with Neural Models. In *Proc. Interspeech 2019*, pages 4190–4194.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Feng Wang, Wei Chen, Zhen Yang, Qianqian Dong, Shuang Xu, and Bo Xu. 2018. Semi-supervised disfluency detection. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3529–3538, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Shaolei Wang, W. Che, and T. Liu. 2016. A neural attention model for disfluency detection. In *COLING*.

W. Wang, G. Tur, J. Zheng, and N. F. Ayan. 2010. Automatic disfluency removal for improving spoken language translation. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5214–5217.

V. Zayats, M. Ostendorf, and H. Hajishirzi. 2014. Multi-domain disfluency and repair detection. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 2907–2911.

Vicky Zayats, Mari Ostendorf, and Hannaneh Hajishirzi. 2016. Disfluency detection using a bidirectional LSTM. *CoRR*, abs/1604.03209.