Autoencoder as Assistant Supervisor: Improving Text Representation for Chinese Social Media Text Summarization

Shuming Ma[1], Xu Sun[1,2], Junyang Lin[3], Houfeng Wang[1]

[1]MOE Key Lab of Computational Linguistics, School of EECS, Peking University
[2]Deep Learning Lab, Beijing Institute of Big Data Research, Peking University
[3]School of Foreign Languages, Peking University
{shumingma, xusun, linjunyang, wanghf}@pku.edu.cn

Abstract

Most of the current abstractive text summarization models are based on the sequence-to-sequence model (Seq2Seq). The source content of social media is long and noisy, so it is difficult for Seq2Seq to learn an accurate semantic representation. Compared with the source content, the annotated summary is short and well written. Moreover, it shares the same meaning as the source content. In this work, we supervise the learning of the representation of the source content with that of the summary. In implementation, we regard a summary autoencoder as an assistant supervisor of Seq2Seq. Following previous work, we evaluate our model on a popular Chinese social media dataset. Experimental results show that our model achieves the state-of-the-art performances on the benchmark dataset.[1]

1 Introduction

Text summarization is to produce a brief summary of the main ideas of the text. Unlike extractive text summarization (Radev et al., 2004; Woodsend and Lapata, 2010; Cheng and Lapata, 2016), which selects words or word phrases from the source texts as the summary, abstractive text summarization learns a semantic representation to generate more human-like summaries. Recently, most models for abstractive text summarization are based on the sequence-to-sequence model, which encodes the source texts into the semantic representation with an encoder, and generates the summaries from the representation with a decoder.

The contents on the social media are long, and contain many errors, which come from spelling mistakes, informal expressions, and grammatical mistakes (Baldwin et al., 2013). Large amount of errors in the contents cause great difficulties for text summarization. As for RNN-based Seq2Seq, it is difficult to compress a long sequence into an accurate representation (Li et al., 2015), because of the gradient vanishing and exploding problem.

Compared with the source content, it is easier to encode the representations of the summaries, which are short and manually selected. Since the source content and the summary share the same points, it is possible to supervise the learning of the semantic representation of the source content with that of the summary.

In this paper, we regard a summary autoencoder as an assistant supervisor of Seq2Seq. First, we train an autoencoder, which inputs and reconstructs the summaries, to obtain a better representation to generate the summaries. Then, we supervise the internal representation of Seq2Seq with that of autoencoder by minimizing the distance between two representations. Finally, we use adversarial learning to enhance the supervision. Following the previous work (Ma et al., 2017), We evaluate our proposed model on a Chinese social media dataset. Experimental results show that our model outperforms the state-of-the-art baseline models. More specifically, our model outperforms the Seq2Seq baseline by the score of 7.1 ROUGE-1, 6.1 ROUGE-2, and 7.0 ROUGE-L.

2 Proposed Model

We introduce our proposed model in detail in this section.

2.1 Notation

Given a summarization dataset that consists of N data samples, the ith data sample (x_i, y_i) con-

---

[1]The code is available at https://github.com/lancopku/superAE
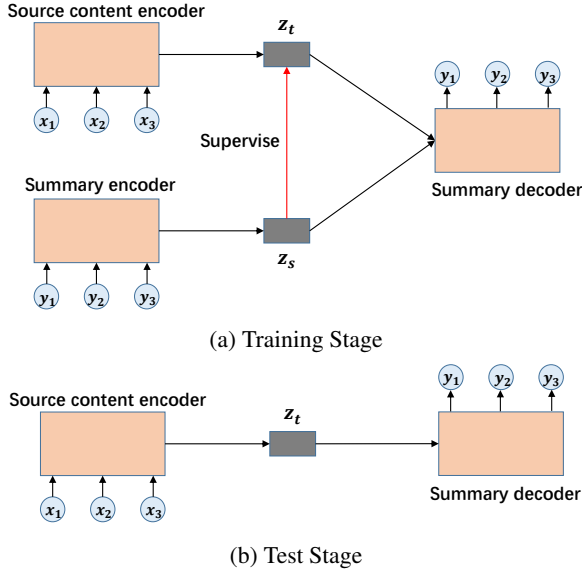
(a) Training Stage

(b) Test Stage

Figure 1: The overview of our model. The model consists of a sequence-to-sequence model and an autoencoder model. At the training stage, we use the autoencoder to supervise the sequence-to-sequence model. At the test stage, we use the sequence-to-sequence model to generate the summaries.

tains a source content $x_i = \{x_1, x_2, ..., x_M\}$, and a summary $y_i = \{y_1, y_2, ..., y_L\}$, while M is the number of the source words, and L is the number of the summary words. At the training stage, we train the model to generate the summary $\boldsymbol{y}$ given the source content $\boldsymbol{x}$. At the test stage, the model decodes the predicted summary $\boldsymbol{y}'$ given the source content $\boldsymbol{x}$.

2.2 Supervision with Autoencoder

Figure 1 shows the architecture of our model. At the training stage, the source content encoder compresses the input contents $\boldsymbol{x}$ into the internal representation z_t with a Bi-LSTM encoder. At the same time, the summary encoder compresses the reference summary $\boldsymbol{y}$ into the representation z_s. Then both z_t and z_s are fed into a LSTM decoder to generate the summary. Finally, the semantic representation of the source content is supervised by the summary.

We implement the supervision by minimizing the distance between the semantic representations z_t and z_s, and this term in the loss function can be written as:

$$L_S = \frac{\lambda}{N_h} d(z_t, z_s) \qquad (1)$$

where $d(z_t, z_s)$ is a function which measures the distance between z_s and z_t. λ is a tunable hyper-parameter to balance the loss of the supervision and the other parts of the loss, and N_h is the number of the hidden unit to limit the magnitude of the distance function. We set $\lambda = 0.3$ based on the performance on the validation set. The distance between two representations can be written as:

$$d(z_t, z_s) = \|z_t - z_s\|_2 \qquad (2)$$

2.3 Adversarial Learning

We further enhance the supervision with the adversarial learning approach. As shown in Eq. 1, we use a fixed hyper-parameter as a weight to measure the strength of the supervision of the autoencoder. However, in the case when the source content and summary have high relevance, the strength of the supervision should be higher, and when the source content and summary has low relevance, the strength should be lower. In order to determine the strength of supervision more dynamically, we introduce the adversarial learning. More specifically, we regard the representation of the autoencoder as the "gold" representation, and that of the sequence-to-sequence as the "fake" representation. A model is trained to discriminate between the gold and fake representations, which is called a discriminator. The discriminator tries to identify the two representations. On the contrary, the supervision, which minimizes the distance of the representations and makes them similar, tries to prevent the discriminator from making correct predictions. In this way, when the discriminator can distinguish the two representations (which means the source content and the summary has low relevance), the strength of supervision will be decreased, and when the discriminator fails to distinguish, the strength of supervision will be improved.

In implementation of the adversarial learning, the discriminator objective function can be written as:

$$\begin{aligned} L_D(\theta_D) = &- \log P_{\theta_D}(y = 1|z_t) \\ &- \log P_{\theta_D}(y = 0|z_s) \end{aligned} \qquad (3)$$

where $P_{\theta_D}(y = 1|z)$ is the probability that the discriminator identifies the vector z as the "gold" representation, while $P_{\theta_D}(y = 0|z)$ is the probability that the vector z is identified as the "fake" representation, and θ_D is the parameters of the discrim-

inator. When minimizing the discriminator objective, we only train the parameters of the discriminator, while the rest of the parameters remains unchanged.

The supervision objective to be against the discriminator can be written as:

$$L_G(\theta_E) = -\log P_{\theta_D}(y = 0|z_t) \\ - \log P_{\theta_D}(y = 1|z_s) \quad (4)$$

When minimizing the supervision objective, we only update the parameters of the encoders.

2.4 Loss Function and Training

There are several parts of the objective functions to optimize in our models. The first part is the cross entropy losses of the sequence-to-sequence and the autoencoder:

$$L_{Seq2seq} = -\sum_{i=1}^{N} p_{Seq2seq}(y_i|z_s) \quad (5)$$

$$L_{AE} = -\sum_{i=1}^{N} p_{AE}(y_i|z_t) \quad (6)$$

The second part is the L2 loss of the supervision, as written in Equation 1. The last part is the adversarial learning, which are Equation 3 and Equation 4. The sum of all these parts is the final loss function to optimize.

We use the Adam (Kingma and Ba, 2014) optimization method to train the model. For the hyper-parameters of Adam optimizer, we set the learning rate $\alpha = 0.001$, two momentum parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$ respectively, and $\epsilon = 1 \times 10^{-8}$. We clip the gradients (Pascanu et al., 2013) to the maximum norm of 10.0.

3 Experiments

Following the previous work (Ma et al., 2017), we evaluate our model on a popular Chinese social media dataset. We first introduce the datasets, evaluation metrics, and experimental details. Then, we compare our model with several state-of-the-art systems.

3.1 Dataset

Large Scale Chinese Social Media Text Summarization Dataset (LCSTS) is constructed by Hu et al. (2015). The dataset consists of more than 2,400,000 text-summary pairs, constructed from a famous Chinese social media website called Sina Weibo.[2] It is split into three parts, with 2,400,591

pairs in PART I, 10,666 pairs in PART II and 1,106 pairs in PART III. All the text-summary pairs in PART II and PART III are manually annotated with relevant scores ranged from 1 to 5. We only reserve pairs with scores no less than 3, leaving 8,685 pairs in PART II and 725 pairs in PART III. Following the previous work (Hu et al., 2015), we use PART I as training set, PART II as validation set, and PART III as test set.

3.2 Evaluation Metric

Our evaluation metric is ROUGE score (Lin and Hovy, 2003), which is popular for summarization evaluation. The metrics compare an automatically produced summary with the reference summaries, by computing overlapping lexical units, including unigram, bigram, trigram, and longest common subsequence (LCS). Following previous work (Rush et al., 2015; Hu et al., 2015), we use ROUGE-1 (unigram), ROUGE-2 (bi-gram) and ROUGE-L (LCS) as the evaluation metrics in the reported experimental results.

3.3 Experimental Details

The vocabularies are extracted from the training sets, and the source contents and the summaries share the same vocabularies. In order to alleviate the risk of word segmentation mistakes, we split the Chinese sentences into characters. We prune the vocabulary size to 4,000, which covers most of the common characters.

We tune the hyper-parameters based on the ROUGE scores on the validation sets. We set the word embedding size and the hidden size to 512, and the number of LSTM layers is 2. The batch size is 64, and we do not use dropout (Srivastava et al., 2014) on this dataset. Following the previous work (Li et al., 2017), we implement the beam search, and set the beam size to 10.

3.4 Baselines

We compare our model with the following state-of-the-art baselines.

- **RNN** and **RNN-cont** are two sequence-to-sequence baseline with GRU encoder and decoder, provided by Hu et al. (2015). The difference between them is that RNN-context has attention mechanism while RNN does not.

- **RNN-dist** (Chen et al., 2016) is a distraction-based neural model, which the attention

Models	R-1	R-2	R-L
RNN-W(Hu et al., 2015)	17.7	8.5	15.8
RNN(Hu et al., 2015)	21.5	8.9	18.6
RNN-cont-W(Hu et al., 2015)	26.8	16.1	24.1
RNN-cont(Hu et al., 2015)	29.9	17.4	27.2
SRB(Ma et al., 2017)	33.3	20.0	30.1
CopyNet-W(Gu et al., 2016)	35.0	22.3	32.0
CopyNet(Gu et al., 2016)	34.4	21.6	31.3
RNN-dist(Chen et al., 2016)	35.2	22.6	32.5
DRGD(Li et al., 2017)	37.0	24.2	34.2
Seq2Seq (our impl.)	32.1	19.9	29.2
+superAE (this paper)	**39.2**	**26.0**	**36.2**
w/o adversarial learning	37.7	25.3	35.2

Table 1: Comparison with state-of-the-art models on the LCSTS test set. R-1, R-2, and R-L denote ROUGE-1, ROUGE-2, and ROUGE-L, respectively. The models with a suffix of 'W' in the table are word-based, while the rest of models are character-based.

mechanism focuses on the different parts of the source content.

- **CopyNet** (Gu et al., 2016) incorporates a copy mechanism to allow parts of the generated summary are copied from the source content.

- **SRB** (Ma et al., 2017) is a sequence-to-sequence based neural model with improving the semantic relevance between the input text and the output summary.

- **DRGD** (Li et al., 2017) is a deep recurrent generative decoder model, combining the decoder with a variational autoencoder.

- **Seq2seq** is our implementation of the sequence-to-sequence model with the attention mechanism, which has the same experimental setting as our model for fair comparison.

3.5 Results

For the purpose of simplicity, we denote our supervision with autoencoder model as **superAE**. We report the ROUGE F1 score of our model and the baseline models on the test sets.

Table 1 summarizes the results of our superAE model and several baselines. We first compare our model with Seq2Seq baseline. It shows that our

Models	2-class (%)	5-class (%)
Seq2seq	80.7	65.1
+superAE	**88.8 (+8.1)**	**71.7 (+6.6)**

Table 2: Accuracy of the sentiment classification on the Amazon dataset. We train a classifier which inputs internal representation provided by the sequence-to-sequence model, and outputs a predicted label. We compute the 2-class and 5-class accuracy of the predicted labels to evaluate the quality of the text representation.

superAE model has a large improvement over the Seq2Seq baseline by 7.1 ROUGE-1, 6.1 ROUGE-2, and 7.0 ROUGE-L, which demonstrates the efficiency of our model. Moreover, we compare our model with the recent summarization systems, which have been evaluated on the same training set and the test sets as ours. Their results are directly reported in the referred articles. It shows that our superAE outperforms all of these models, with a relative gain of 2.2 ROUGE-1, 1.8 ROUGE-2, and 2.0 ROUGE-L. We also perform ablation study by removing the adversarial learning component, in order to show its contribution. It shows that the adversarial learning improves the performance of 1.5 ROUGE-1, 0.7 ROUGE-2, and 1.0 ROUGE-L.

We also give a summarization examples of our model. As shown in Table 3, the SeqSeq model captures the wrong meaning of the source content, and produces the summary that "*China United Airlines exploded in the airport*". Our superAE model captures the correct points, so that the generated summary is close in meaning to the reference summary.

3.6 Analysis of text representation

We want to analyze whether the internal text representation is improved by our superAE model. Since the text representation is abstractive and hard to evaluate, we translate the representation into a sentiment score with a sentiment classifier, and evaluate the quality of the representation by means of the sentiment accuracy.

We perform experiments on the Amazon Fine Foods Reviews Corpus (McAuley and Leskovec, 2013). The Amazon dataset contains users' rating labels as well as the summary for the reviews, making it possible to train a classifier to predict the sentiment labels and a seq2seq model to generate summaries. First, we train the superAE model and

Source: 昨晚，中联航空成都飞北京一架航班被发现有多人吸烟。后因天气原因，飞机备降太原机场。有乘客要求重新安检，机长决定继续飞行，引起机组人员与未吸烟乘客冲突。

Last night, several people were caught to smoke on a flight of China United Airlines from Chendu to Beijing. Later the flight temporarily landed on Taiyuan Airport. Some passengers asked for a security check but were denied by the captain, which led to a collision between crew and passengers.

Reference: 航班多人吸烟机组人员与乘客冲突。

Several people smoked on a flight which led to a collision between crew and passengers.

Seq2Seq: 中联航空机场发生爆炸致多人死亡。

China United Airlines exploded in the airport, leaving several people dead.

+superAE: 成都飞北京航班多人吸烟机组人员与乘客冲突。

Several people smoked on a flight from Chendu to Beijing, which led to a collision between crew and passengers.

Table 3: A summarization example of our model, compared with Seq2Seq and the reference.

the seq2seq model with the text-summary pairs until convergence. Then, we transfer the encoders to a sentiment classifier, and train the classifier with fixing the parameters of the encoders. The classifier is a simple feedforward neural network which maps the representation into the label distribution. Finally, we compute the accuracy of the predicted 2-class labels and 5-class labels.

As shown in Table 2, the seq2seq model achieves 80.7% and 65.1% accuracy of 2-class and 5-class, respectively. Our superAE model outperforms the baselines with a large margin of 8.1% and 6.6%.

4 Related Work

Rush et al. (2015) first propose an abstractive based summarization model, which uses an attentive CNN encoder to compress texts and a neural network language model to generate summaries. Chopra et al. (2016) explore a recurrent structure for abstractive summarization. To deal with out-of-vocabulary problem, Nallapati et al. (2016)

propose a generator-pointer model so that the decoder is able to generate words in source texts. Gu et al. (2016) also solved this issue by incorporating copying mechanism, allowing parts of the summaries are copied from the source contents. See et al. (2017) further discuss this problem, and incorporate the pointer-generator model with the coverage mechanism. Hu et al. (2015) build a large corpus of Chinese social media short text summarization, which is one of our benchmark datasets. Chen et al. (2016) introduce a distraction based neural model, which forces the attention mechanism to focus on the difference parts of the source inputs. Ma et al. (2017) propose a neural model to improve the semantic relevance between the source contents and the summaries.

Our work is also related to the sequence-to-sequence model (Cho et al., 2014), and the autoencoder model (Bengio, 2009; Liou et al., 2008, 2014). Sequence-to-sequence model is one of the most successful generative neural model, and is widely applied in machine translation (Sutskever et al., 2014; Jean et al., 2015; Luong et al., 2015), text summarization (Rush et al., 2015; Chopra et al., 2016; Nallapati et al., 2016), and other natural language processing tasks. Autoencoder (Bengio, 2009) is an artificial neural network used for unsupervised learning of efficient representation. Neural attention model is first proposed by Bahdanau et al. (2014).

5 Conclusion

We propose a novel model, in which the autoencoder is a supervisor of the sequence-to-sequence model, to learn a better internal representation for abstractive summarization. An adversarial learning approach is introduced to further improve the supervision of the autoencoder. Experimental results show that our model outperforms the sequence-to-sequence baseline by a large margin, and achieves the state-of-the-art performances on a Chinese social media dataset.

Acknowledgements

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Timothy Baldwin, Paul Cook, Marco Lui, Andrew MacKinlay, and Li Wang. 2013. How noisy social media text, how diffrnt social media sources? In *Sixth International Joint Conference on Natural Language Processing, IJCNLP 2013, Nagoya, Japan, October 14-18, 2013*, pages 356–364.

Yoshua Bengio. 2009. Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, 2(1):1–127.

Qian Chen, Xiaodan Zhu, Zhenhua Ling, Si Wei, and Hui Jiang. 2016. Distraction-based neural networks for modeling documents. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI 2015)*, New York, NY. AAAI.

Jianpeng Cheng and Mirella Lapata. 2016. Neural summarization by extracting sentences and words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014*, pages 1724–1734.

Sumit Chopra, Michael Auli, and Alexander M. Rush. 2016. Abstractive sentence summarization with attentive recurrent neural networks. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O. K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016*.

Baotian Hu, Qingcai Chen, and Fangze Zhu. 2015. LCSTS: A large scale chinese short text summarization dataset. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1967–1972.

Sébastien Jean, KyungHyun Cho, Roland Memisevic, and Yoshua Bengio. 2015. On using very large target vocabulary for neural machine translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics, ACL 2015*, pages 1–10.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Jiwei Li, Minh-Thang Luong, and Dan Jurafsky. 2015. A hierarchical neural autoencoder for paragraphs and documents. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1106–1115.

Piji Li, Wai Lam, Lidong Bing, and Zihao Wang. 2017. Deep recurrent generative decoder for abstractive text summarization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 2091–2100.

Chin-Yew Lin and Eduard H. Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, HLT-NAACL 2003*.

Cheng-Yuan Liou, Wei-Chen Cheng, Jiun-Wei Liou, and Daw-Ran Liou. 2014. Autoencoder for words. *Neurocomputing*, 139:84–96.

Cheng-Yuan Liou, Jau-Chi Huang, and Wen-Chie Yang. 2008. Modeling word perception using the elman network. *Neurocomputing*, 71(16-18):3150–3157.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015*, pages 1412–1421.

Shuming Ma, Xu Sun, Wei Li, Sujian Li, Wenjie Li, and Xuancheng Ren. 2018. Query and output: Generating words by querying distributed word representations for paraphrase generation. In *NAACL 2018*.

Shuming Ma, Xu Sun, Jingjing Xu, Houfeng Wang, Wenjie Li, and Qi Su. 2017. Improving semantic relevance for sequence-to-sequence learning of chinese social media text summarization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pages 635–640.

Julian John McAuley and Jure Leskovec. 2013. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. In *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, pages 897–908.

Ramesh Nallapati, Bowen Zhou, Cícero Nogueira dos Santos, Çaglar Gülçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, August 11-12, 2016*, pages 280–290.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, pages 1310–1318.

Dragomir R. Radev, Timothy Allison, Sasha Blair-Goldensohn, John Blitzer, Arda Çelebi, Stanko Dimitrov, Elliott Drábek, Ali Hakim, Wai Lam, Danyu Liu, Jahna Otterbacher, Hong Qi, Horacio Saggion, Simone Teufel, Michael Topper, Adam Winkel, and Zhu Zhang. 2004. MEAD - A platform for multidocument multilingual text summarization. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation, LREC 2004*.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 379–389.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017*, pages 1073–1083.

Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Xu Sun, Xuancheng Ren, Shuming Ma, and Houfeng Wang. 2017a. meprop: Sparsified back propagation for accelerated deep learning with reduced overfitting. In *ICML 2017*, pages 3299–3308.

Xu Sun, Bingzhen Wei, Xuancheng Ren, and Shuming Ma. 2017b. Label embedding network: Learning label representation for soft training of deep networks. *CoRR*, abs/1710.10393.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014*, pages 3104–3112.

Sho Takase, Jun Suzuki, Naoaki Okazaki, Tsutomu Hirao, and Masaaki Nagata. 2016. Neural headline generation on abstract meaning representation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1054–1059.

Kristian Woodsend and Mirella Lapata. 2010. Automatic generation of story highlights. In *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 565–574.

Jingjing Xu, Xu Sun, Xuancheng Ren, Junyang Lin, Binzhen Wei, and Wei Li. 2018a. Dp-gan: Diversity-promoting generative adversarial network for generating informative and diversified text. *CoRR*, abs/1802.01345.

Jingjing Xu, Xu Sun, Qi Zeng, Xiaodong Zhang, Xuancheng Ren, Houfeng Wang, and Wenjie Li. 2018b. Unpaired sentiment-to-sentiment translation: A cycled reinforcement learning approach. In *ACL 2018*.