Unsupervised Neural Text Simplification

[1]Sai Surya, [2]Abhijit Mishra, [2]Anirban Laha, [2]Parag Jain, [2]Karthik Sankaranarayanan

[1]IIT Kharagpur
[2]IBM Research India
subramanyamdvss@gmail.com
{abhijimi,anirlaha,pajain34,kartsank}@in.ibm.com

ACL 2019

Github repository

Overview

- **Task:** Text Simplification has numerous use-cases in education technology, targeted content creation and language learning.
- **Background:** Data driven simplification requires costly parallel simplification pairs, moreover current public datasets on simplification have been prone to noise. (Coster and Kauchak, 2011).
- **Objective:** We aim to make use of unlabeled corpora of simple and complex sentences to learn simplification knowledge.
- **Results:** Our analysis on public test data shows that the proposed model can perform text-simplification at both lexical and syntactic levels, competitive to existing supervised methods and unsupervised methods.

Datasets & Model Architecture

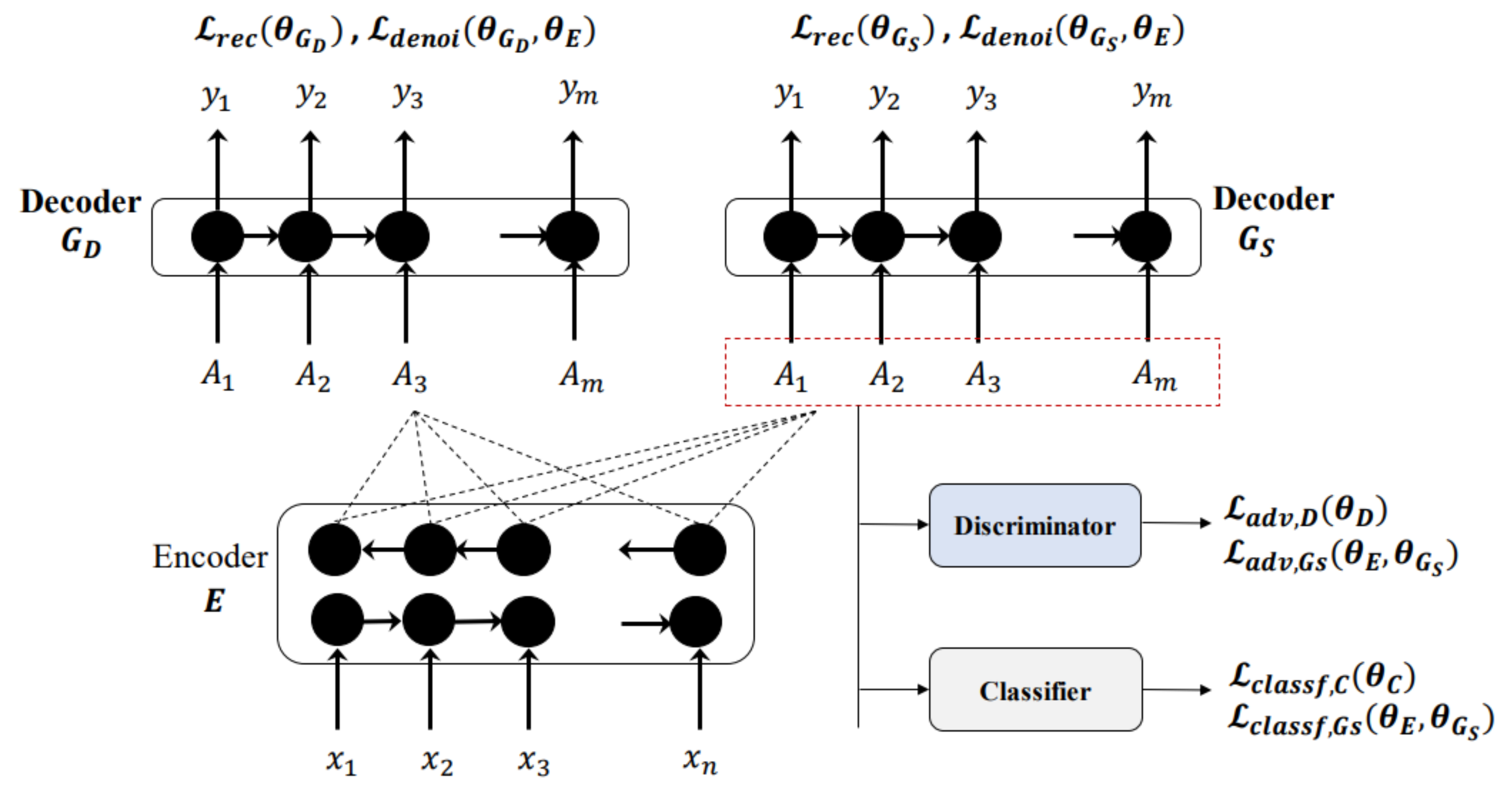
Datasets:

- An unlabeled dataset of simple and complex sentences judiciously by partitioning the *standard en-wikipedia dump*, using ***readability metrics***

Category	#Sents	Avg. Words	Avg. FE	FE-Range
Simple	720k	18.23	76.67	74.9-79.16
Complex	720k	35.03	7.26	5.66-9.93

Architecture:

- Built based on the encode-attend-decode architecture
- Encoder E, Decoders G_s and G_d use layers of GRUs
- Discriminator and Classifier enforce losses on attention vectors such that G_s generates simple sentences, given any input at encoder.



Training Scheme

Training Losses:

- **Reconstruction Loss**: E-G_s is trained to reconstruct simple sentences and E-G_d is trained to reconstruct difficult sentences
- **Adversarial Loss:** Distribution of Context vectors extracted by G_s from a complex sentence should resemble the context vectors from a simple sentence.
- **Diversification Loss**: This helps E-G_s to learn to generate simple context vectors distinguishable from complex context vectors.
- **Denoising Loss**: Denoising is helpful to learn syntactic/structural transformations.

Algorithm 1 Unsupervised simplification algorithm using denoising, reconstruction, adversarial and diversification losses.

Input: simple dataset $\mathcal{S}$, complex dataset $\mathcal{D}$.

Initialization phase:
repeat
  Update $\theta_E, \theta_{G_s}, \theta_{G_d}$ using $\mathcal{L}_{denoi}$
  Update $\theta_E, \theta_{G_s}, \theta_{G_d}$ using $\mathcal{L}_{rec}$
  Update θ_D, θ_C using $\mathcal{L}_{adv,D}$ $\mathcal{L}_{div,C}$
until specified number of steps are completed
Adversarial phase:
repeat
  Update $\theta_E, \theta_{G_s}, \theta_{G_d}$ using $\mathcal{L}_{denoi}$
  Update $\theta_E, \theta_{G_s}, \theta_{G_d}$ using $\mathcal{L}_{adv,G_s}$, $\mathcal{L}_{div,G_s}$, $\mathcal{L}_{rec}$
  Update θ_D, θ_C using $\mathcal{L}_{adv,D}$, $\mathcal{L}_{div,C}$
until specified number of steps are completed

Unsupervised Training

Algorithm 2 Semi-supervised simplification algorithm using denoising, reconstruction, adversarial and diversification losses followed by cross-entropy loss using parallel data.

Input: simple dataset $\mathcal{S}$, complex dataset $\mathcal{D}$, parallel dataset $\Delta = (\mathcal{S}_p, \mathcal{D}_p)$

Initialization phase:
repeat
  Update $\theta_E, \theta_{G_s}, \theta_{G_d}$ using $\mathcal{L}_{denoi}$
  Update $\theta_E, \theta_{G_s}, \theta_{G_d}$ using $\mathcal{L}_{rec}$
  Update θ_D, θ_C using $\mathcal{L}_{adv,D}$ $\mathcal{L}_{div,C}$
until specified number of steps are completed
Adversarial phase:
repeat
  Update $\theta_E, \theta_{G_s}, \theta_{G_d}$ using $\mathcal{L}_{denoi}$
  Update $\theta_E, \theta_{G_s}, \theta_{G_d}$ using $\mathcal{L}_{adv,G_s}$, $\mathcal{L}_{div,G_s}$, $\mathcal{L}_{rec}$
  Update θ_D, θ_C using $\mathcal{L}_{adv,D}$, $\mathcal{L}_{div,C}$
  Update θ_E, θ_{G_s} using $\mathcal{L}_{cross}$
  Update θ_E, θ_{G_d} using $\mathcal{L}_{cross}$
until specified number of steps are completed

Semi-supervised Training

Experiment & Results

- Comparisons with supervised systems - NTS, SBMT, PBSMT, unsupervised systems – UNMT, USMT, ST.
- Both automatic and human evaluations are used to compare with existing baselines.

System	FE-diff	SARI	BLEU	Word-diff
UNTS+10K	10.45	35.29	76.13	2.38
UNTS	11.15	33.8	74.24	**3.55**
UNMT	6.60	33.72	70.84	0.74
USMT	13.84	32.11	87.36	-0.01
ST	54.38	14.97	0.73	5.61
NTS	5.37	36.1	79.38	2.73
SBMT	**17.68**	**38.59**	73.62	-0.84
PBSMT	9.14	34.07	67.79	2.26
LIGHTLS	3.01	34.96	**83.54**	-0.02

Automatic Evaluation

System	Simpleness	Fluency	Relatedness
UNTS+10K	**57%**	4.13	3.93
UNTS	47%	3.86	3.73
UNMT	40%	3.8	**4.06**
NTS	49%	4.13	3.26
SBMT	53%	**4.26**	**4.06**
PBSMT	53%	3.8	3.93
LIGHTLS	6%	4.2	3.33

Manual Evaluation

Analysis & Conclusion

Types of simplifications generated by the model:

Type of Simplification	Source	Prediction
Splitting	Calvin Baker is an American novelist .	Calvin Baker is an American . American Baker is a birthplace .
Sentence Shortening	During an interview , Edward Gorey mentioned that Bawden was one of his favorite artists , **lamenting the fact that not many people remembered or knew about this fine artist .**	During an interview , Edward Gorey mentioned that Bawden was one of his favorite artists .
Lexical Replacement	In **architectural** decoration Small pieces of colored and **iridescent** shell have been used to create **mosaics** and inlays , which have been used to decorate walls , furniture and boxes .	In **impressive** decoration Small pieces of colored and **reddish** shell have been used to create **statues** and inlays , which have been used to decorate walls , furniture and boxes .

Generated example outputs

Conclusion:

- First attempt towards unsupervised neural text simplification that relies only on unlabeled text corpora.
- Judicious selection of training corpora through readability
- In future, would like to incorporate training schemes to tackle complex syntactic simplification operations.

References:

- Sergiu Nisioi, Sanja Stajner, Simone Paolo Ponzetto and Liviu P Dinu. 2017. Exploring neural text simplification models. In ACL, volume 2, pages 85–91.
- Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018b. Unsupervised neural machine translation. In Proceedings of the Sixth International Conference on Learning Representations
- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. arXiv:1409.0473.
- William Coster and David Kauchak. 2011. Simple english wikipedia: a new text simplification task. In ACL, pages 665–669. Association for Computational Linguistics.
- Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014b. 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), pages 1724– 1734.

IBM Research AI