Data Augmentation Techniques through Transformers for Text Classification

José M. Tapia Téllez

Coordinación de Ciencias Computacionales
Instituto Nacional de Astrofísica, Óptica y Electrónica

Machine Learning II

Content

Introduction

Motivation

- Increase the size of the training data
  - Reduce overfitting
  - Enhance robustness of the the ML model in the case of low data
- Text classification is a fundamental task in NLP
  - Create and explore language transformation rules to come up with generalized transformation rules

Problems

- Methods without ML are based on synonyms and lack to include the context of the sentence

Introduction

Problems

- It is not easy to come up with generalized rules for language transformation
- Preserving the class labels

Solution

- Pretrained Bert model in order to mask and augment data.
  - Single masking (one, three, five and ten)
  - Double masking
  - Triple masking
- Pretrained GPT2 for sentence augmentation and thus augment the data

Related Work

EDA (easy data augmentation), proposed in Wei & Zour (2019)

- Four simple operations: synonym replacement, random insertion, random swap, and random deletion.
- Improves performance in five different text classification on RNN and CNN (Better results on smaller datasets.)

Contextual Augmentation, proposed in Kobayashi (2018)

- They proposed contextual augmentation, which replaces a word based on a bi-directional language model at the word positions.
- It also augments the number of sentences based in the label-conditional architecture.

Related Work

Conditional Data Augmentation proposed in Kumar et al. (2020)

- They study different types of pre-trained transformer based models such as GPT-2, BERT and BART for conditional data augmentation.
- On three classification benchmarks, pre-trained Seq2Seq model outperforms other models
- They're method is a nice approach into the study of class-label information.

Background

Transformers

- Network architecture based on attention mechanisms
- Provides general-purpose architectures (BERT, GPT-2 for NLU and NLG).

Bidirectional Encoder Representations from Transformers

- BERT is a Language Representation Model designed to pre-train deep bidirectional representation from unlabeled text by jointly conditioning on both left and right context in all layers.
- Bert was trained with a masked language modelling (MLM) objective. Therefore it is efficient predicting masked tokens.

Background

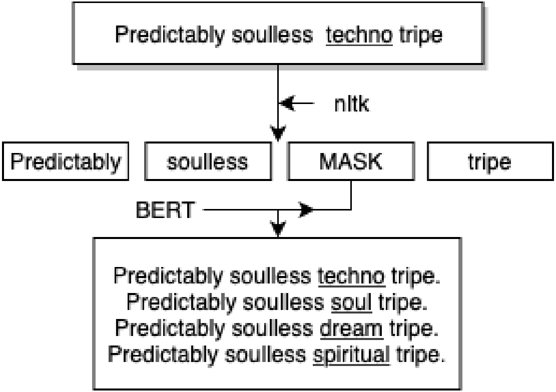
GPT-2

- Causal unidirectional transformer pre-trained using language modeling on a very large corpus of 40GB of text data.
- GPT-2 is trained with a simple objective: predict the next word, given all of the previous words in some text

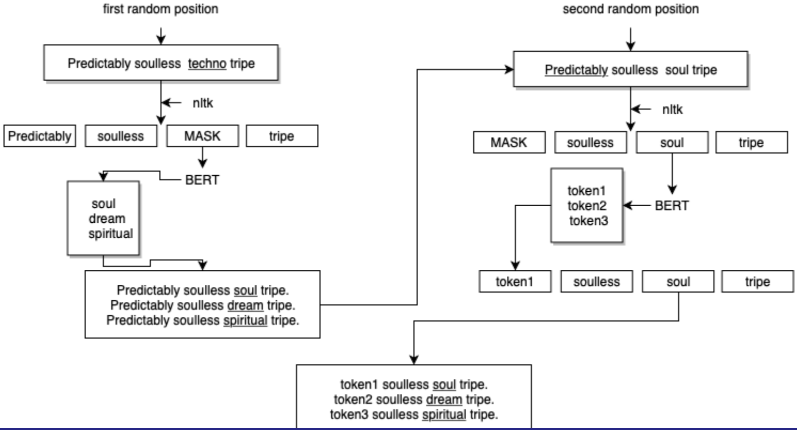
Natural Language Toolkit (NLTK)

- Suite of open source program modules providing ready-to-use computational linguistics courseware.

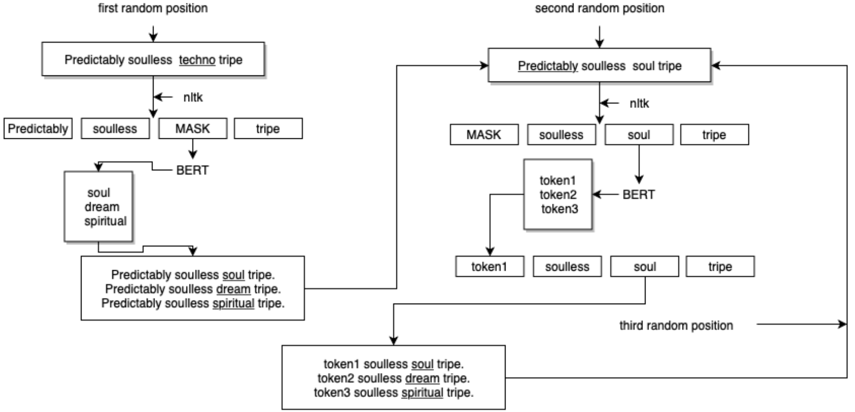
Single Masking Augmentation



Double Masking Augmentation



Triple Masking Augmentation



Technical Features

BERT

- Libraries needed: BertTokenizer, BertForMaskedLM
- Model: bert-base-uncased

Sentence Augmentation

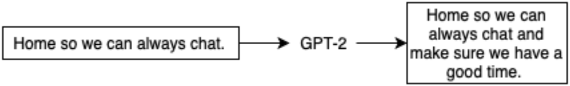


Figura: GPT-2 Scheme-example for Sentence Augmentation

Technical Features

- Transformer Libraries: TFGPT2LMHeadModel, GPT2 Tokenizer
- Model: gpt2

Experiments Specifications

RNN

- Input Layer, Embeddings, LSTM, FC1 (Dense), Activation (relu), Dropout, Output Layer, Activation (sigmoid)
- Compilation: loss = 'binary_crossentropy', optimizer = RMSprop
- Fit: batch size = 128, epochs = 10

CNN

- Embedding, Conv1D (activation = relu), Dropout (0.5), Global Max Pooling 1D, Dense (relu), Dense(sigmoid)
- Compilation: optimizer = 'adam', loss = 'binary_crossentropy'
- Fit: epochs = 10

Frame Title

Datasets

- SST-2: ALBERT → 97.4 %
- Sentence Type: BERT → 65.5 %
- IMDB: XLNET → 96.800 %
- Spam: ELCADP → 95.1 %

Single Masking

		Plain	Plain + Single Masking			
			1	3	5	10
Spam	RNN	0.79	0.86	0.84	0.86	0.92
	CNN	0.59	0.68	0.67	0.65	0.63
IMDB	RNN	0.71	0.72	0.77	0.75	0.71
	CNN	0.79	0.76	0.77	0.72	0.74
Sentence Type	RNN	0.35	0.42	0.46	0.53	0.56
	CNN	0.40	0.43	0.44	0.40	0.40
SST2	RNN	0.59	0.63	0.68	0.66	0.67
	CNN	0.68	0.63	0.66	0.65	0.71
Average	RNN	0.61	0.65	0.68	0.70	0.71
	CNN	0.61	0.62	0.63	0.60	0.62

Double Masking Augmentation

		Plain	Plain + Double Masking 1	3	5	10
Spam	RNN	0.79	0.82	0.77	0.78	0.86
	CNN	0.59	0.65	0.57	0.66	0.74
IMDB	RNN	0.71	0.71	0.69	0.76	0.70
	CNN	0.79	0.76	0.76	0.74	0.70
Sentence Type	RNN	0.35	0.37	0.44	0.35	0.50
	CNN	0.40	0.43	0.40	0.35	0.50
SST2	RNN	0.59	0.64	0.66	0.69	0.66
	CNN	0.68	0.68	0.65	0.70	0.70
Average	RNN	0.61	0.63	0.64	0.64	0.68
	CNN	0.61	0.63	0.59	0.61	0.66

Triple Masking Augmentation

		Plain	Plain + Triple Masking			
			1	3	5	10
Spam	RNN	0.79	0.74	0.77	0.67	0.85
	CNN	0.59	0.69	0.65	0.62	0.57
IMDB	RNN	0.71	0.72	0.70	0.70	0.77
	CNN	0.79	0.75	0.76	0.70	0.76
Sentence Type	RNN	0.35	0.43	0.38	0.47	0.51
	CNN	0.40	0.42	0.35	0.36	0.50
SST2	RNN	0.59	0.64	0.69	0.70	0.63
	CNN	0.68	0.70	0.71	0.70	0.63
Average	RNN	0.61	0.63	0.63	0.63	0.69
	CNN	0.61	0.64	0.62	0.59	0.61

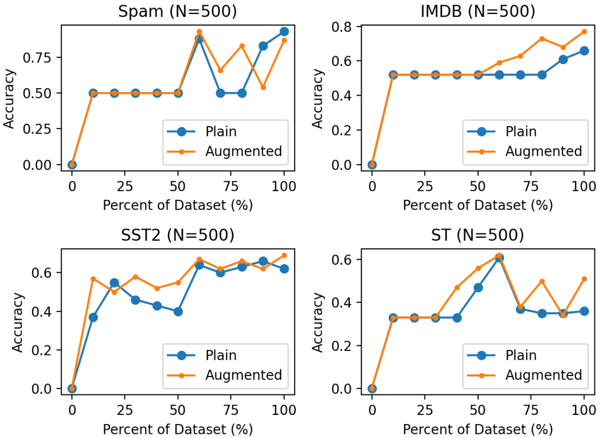
Augmented Sentence

		Plain	Plain + Augmented Sentence
Spam	RNN	0.79	0.75
	CNN	0.59	0.61
IMDB	RNN	0.71	0.75
	CNN	0.79	0.76
Sentence Type	RNN	0.35	0.46
	CNN	0.40	0.41
SST2	RNN	0.59	0.62
	CNN	0.68	0.67
Average	RNN	0.61	0.64
	CNN	0.61	0.61

Augmented Techniques Compared

		RNN	CNN
Plain		0.61	0.61
Plain + Single Masking	1	0.65	0.62
	3	0.68	0.63
	5	0.70	0.60
	10	0.71	0.62
Plain + Double Masking	1	0.63	0.63
	3	0.64	0.59
	5	0.64	0.61
	10	0.68	0.66
Plain + Triple Masking	1	0.63	0.64
	3	0.63	0.62
	5	0.63	0.59
	10	0.69	0.61
Augmented Sentence		0.64	0.61

Graphs


Spam (N=500)
IMDB (N=500)
SST2 (N=500)
ST (N=500)
Accuracy
Percent of Dataset (%)
Plain
Augmented
0.00
0.25
0.50
0.75
0.0
0.2
0.4
0.6
0.8
0
25
50
75
100

Conclusions

- We were able to create DA techniques based on Transformers
- On average, augmented data by single masking obtains better accuracy results than plain data on RNN and CNN.
- On average, augmented data by double masking obtain better accuracy results than plain data on RNN and CNN
- On average, augmented data by triple masking obtains better accuracy results that plain data on RNN and CNN
- On average, augmented data by augmented sentence obtain better accuracy results than plain data on RNN.
- Based on the graph, the best method (on average) gets the best results in approximately 25 to 75 % of the data
- In Augmented Sentence, augment different sentence with differente lengths (Too slow to work with)

Future Work

- Work with bigger data and through this, be able to make average comparisons.
- Bigger data with also help get a better graph representations
- Include in our methods a class label identifier as in Kobayashi (2018) and Kumar et al. (2020).

Bibliography

Kobayashi, S. (2018). Contextual augmentation: Data augmentation by words with paradigmatic relations. *arXiv preprint arXiv:1805.06201*.

Wei, J., & Zou, K. (2019). Eda: Easy data augmentation techniques for boosting performance obtain text classification tasks. *arXiv preprint arXiv:1901.11196*.

Kumar, V., Choudhary, A., & Cho, E. (2020). Data augmentation using pre-trained transformer models. *arXiv preprint arXiv:2003.02245*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems (pp. 5998-6008)*.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8), 9.

Loper, E., & Bird, S. (2002). NLTK: the natural language toolkit. *arXiv preprint cs/0205028*.

"Papers with Code - *The Latest in Machine Learning.*" *The Latest in Machine Learning* — Papers With Code, paperswithcode.com/.