Document Classification Using BERT

Hussam Kaka

Resources

DocBERT: BERT for Document Classification

Ashutosh Adhikari, Achyudh Ram, Raphael Tang, and **Jimmy Lin**
David R. Cheriton School of Computer Science
University of Waterloo
{adadhika, arkeshav, r33tang, jimmylin}@uwaterloo.ca

Abstract

We present, to our knowledge, the first application of BERT to document classification. A few characteristics of the task might lead one to think that BERT is not the most appropriate model: syntactic structures matter less

an unsupervised objective of masked languag modeling and next-sentence prediction. Next, thi pre-trained network is then fine-tuned on task specific, labeled data.

BERT, however, has not yet been fine-tuned fo document classification. Why is this worth ex

How to Fine-Tune BERT for Text Classification?

Chi Sun, Xipeng Qiu[*], Yige Xu, Xuanjing Huang
Shanghai Key Laboratory of Intelligent Information Processing, Fudan University
School of Computer Science, Fudan University
825 Zhangheng Road, Shanghai, China
{sunc17,xpqiu,ygxu18,xjhuang}@fudan.edu.cn

Abstract

Language model pre-training has proven to be useful in learning universal language representations. As a state-of-the-art language model pre-training model, BERT (Bidirectional En-

2018). These word embeddings are often used as additional features for the main task. Another kind of pre-training models is sentence-level. Howard and Ruder (2018) propose ULM-FiT, a fine-tuning method for pre-trained language

Classification at a Glance

- Many applications
  - Sentiment analysis, text tagging, spam detection, intent detection
- Widely studied problem
  - Results available on many dataset
  - Easy to compare performance to prior literature
- High results are achievable on publicly available datasets
- Previous models concentrated on neural architecture, with inputs from pre-trained word embeddings (e.g. LSTM).

Why use BERT for classification?

- Recall that BERT
  - Is pre-trained (unsupervised) on a large corpus of text
  - Uses a transformer model (12 or 28)
  - Fine tuned on specific task
- BERT has achieved state of the art results
  - In question answering (SQuAD)
  - A variety of NLP tasks including sentence classification (GLUE)
- Possibility of reduced task-specific training given transfer learning

BERT Challenges for Doc Classification

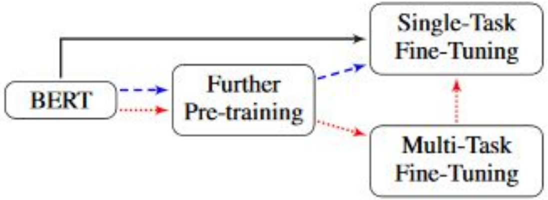
- Computational expense
  - Hundreds of millions of parameters
  - High memory requirements
  - Inference is also computationally expensive
- Pre-training is not domain specific
  - Would classification of medical records require pre-training on a large corpus of medical data? (BioBERT)
- Input length is limited to 512 tokens
  - What about longer documents?

Semantics

- **Pre-training:** unsupervised training by feeding text to BERT.
    - BERT learns by masking words and trying to predict them.
    - A pre-trained BERT model can be further pre-trained.
- **Fine-tuning:** supervised learning by feeding text with a label to BERT.
    - Minimize cross entropy.

Further pre-training

- Within-task pre-training
  - Dataset from target task
- In-domain pre-training
  - Different datasets from same domain as task
- Cross-domain pre-training
  - As the name implies.



Fine tuning of BERT

- BERT takes the final hidden state of the first token ([CLS]) as a representation of the whole text.
- Add softmax layer to output $$p(c|\mathbf{h}) = softmax(W|\mathbf{h})$$

$$W \in \mathbb{R}^{K \times H}$$

- Train the entire model, BERT + softmax layer, using cross entropy or binary cross entropy.

Datasets: Sun et al.

Dataset	Classes	Type	Average lengths	Max lengths	Exceeding ratio	Train samples	Test samples
IMDb	2	Sentiment	292	3,045	12.69%	25,000	25,000
Yelp P.	2	Sentiment	177	2,066	4.60%	560,000	38,000
Yelp F.	5	Sentiment	179	2,342	4.60%	650,000	50,000
TREC	6	Question	11	39	0.00%	5,452	500
Yahoo! Answers	10	Question	131	4,018	2.65%	1,400,000	60,000
AG's News	4	Topic	44	221	0.00%	120,000	7,600
DBPedia	14	Topic	67	3,841	0.00%	560,000	70,000
Sogou News	6	Topic	737	47,988	46.23%	54,000	6,000

Datasets: Adhikari et al.

Dataset	C	N	W	S
Reuters	90	10,789	144.3	6.6
AAPD	54	55,840	167.3	1.0
IMDB	10	135,669	393.8	14.4
Yelp 2014	5	1,125,386	148.8	9.1

Challenges of fine tuning

1. Overcoming max document length
    - BERT takes maximum input length of 512
    - Must start with a [CLS] token and end with a [SEP] token
2. Selecting the best BERT layer for classification
    - First layer? Deepest? Somewhere in between?
3. Choosing an optimizer to minimize over-fitting

1. len(document) > 512

- Truncation methods
  - **Head**: first 510 tokens
  - **Tail**: last 510 tokens
  - **Head+Tail**: first 128 and last 382 tokens
- Hierarchical methods
  - Divide text L into L/510 fractions
  - Mean pooling, max pooling and self attention to combine hidden states of [CLS] for each fraction
- Adhikari et. al do not address this issue

Method	IMDb	Sogou
head-only	5.63	2.58
tail-only	5.44	3.17
head+tail	**5.42**	**2.43**
hier. mean	5.89	2.83
hier. max	5.71	2.47
hier. self-attention	5.49	2.65

Test error rates. IMDb and Chinese Sogou News.

Take away points

1. Document length problem can be overcome.

2. Selecting the best layer for classification

- First layer may learn more general information
- Deepest layer may contain most high level information

Layer	Test error rates(%)
Layer-0	11.07
Layer-1	9.81
Layer-2	9.29
Layer-3	8.66
Layer-4	7.83
Layer-5	6.83
Layer-6	6.83
Layer-7	6.41
Layer-8	6.04
Layer-9	5.70
Layer-10	5.46
Layer-11	**5.42**
First 4 Layers + concat	8.69
First 4 Layers + mean	9.09
First 4 Layers + max	8.76
Last 4 Layers + concat	5.43
Last 4 Layers + mean	5.44
Last 4 Layers + max	**5.42**
All 12 Layers + concat	5.44

Conclusion: use deepest layer.

3. An optimizer to minimize over-fitting

- Hypothesis: giving smaller learning rates to lower layers improves performance
- Decrease learning rates by a decay factor

$$\eta^{k-1} = \xi \cdot \eta^{k}$$

- Note: this is for fine-tuning a pre-trained model.

Learning rate	Decay factor ξ	Test error rates(%)
2.5e-5	1.00	5.52
2.5e-5	0.95	5.46
2.5e-5	0.90	**5.44**
2.5e-5	0.85	5.58
2.0e-5	1.00	5.42
2.0e-5	0.95	**5.40**
2.0e-5	0.90	5.52
2.0e-5	0.85	5.65

Conclusion: a decay factor improves performance slightly.

Take away points

1. Document length problem can be overcome.

2. Use a decay factor for layer learning rates.

Results

Model	IMDb	Yelp P.	Yelp F.	TREC	Yah. A.	AG	DBP	Sogou	Avg. Δ
BERT-Feat	6.79	2.39	30.47	4.20	22.72	5.92	0.70	2.50	-
BERT-FiT	5.40	2.28	30.06	2.80	22.42	5.25	0.71	2.43	9.22%
BERT-ITPT-FiT	**4.37**	1.92	29.42	3.20	22.38	**4.80**	0.68	**1.93**	16.07%
BERT-IDPT-FiT	4.88	**1.87**	29.25	**2.20**	**21.86**	4.88	**0.65**	/	**18.57%**
BERT-CDPT-FiT	5.18	1.97	**29.20**	2.80	21.94	5.08	0.67	/	14.38%

Feat: BERT as features
FiT: fine tuning
ITPT: within-task pre-training
IDPT: within-domain pre-training
CDPT: cross-domain pre-training

Comparison to prior models

Model	IMDb	Yelp P.	Yelp F.	TREC	Yah. A.	AG	DBP	Sogou	Avg. Δ
Char-level CNN(Zhang et al., 2015)	/	4.88	37.95	/	28.80	9.51	1.55	3.80*	/
VDCNN (Conneau et al., 2016)	/	4.28	35.28	/	26.57	8.67	1.29	3.28	/
DPCNN (Johnson and Zhang, 2017)	/	2.64	30.58	/	23.90	6.87	0.88	3.48*	/
D-LSTM (Yogatama et al., 2017)	/	7.40	40.40	/	26.30	7.90	1.30	5.10	/
Standard LSTM (Seo et al., 2017)	8.90	/	/	/	/	6.50	/	/	/
Skim-LSTM (Seo et al., 2017)	8.80	/	/	/	/	6.40	/	/	/
HAN (Yang et al., 2016)	/	/	/	/	24.20	/	/	/	/
Region Emb. (Qiao et al., 2018)	/	3.60	35.10	/	26.30	7.20	1.10	2.40	/
CoVe (McCann et al., 2017)	8.20	/	/	4.20	/	/	/	/	/
ULMFiT (Howard and Ruder, 2018)	4.60	2.16	29.98	3.60	/	5.01	0.80	/	/
BERT-Feat	6.79	2.39	30.47	4.20	22.72	5.92	0.70	2.50	-
BERT-FiT	5.40	2.28	30.06	2.80	22.42	5.25	0.71	2.43	9.22%
BERT-ITPT-FiT	**4.37**	1.92	29.42	3.20	22.38	**4.80**	0.68	**1.93**	16.07%
BERT-IDPT-FiT	4.88	**1.87**	29.25	**2.20**	**21.86**	4.88	**0.65**	/	**18.57%**
BERT-CDPT-FiT	5.18	1.97	**29.20**	2.80	21.94	5.08	0.67	/	14.38%

Conclusion: BERT scores best on all datasets

BERT large vs BERT base

Model	IMDb	Yelp P.	Yelp F.	AG	DBP
ULMFiT	4.60	2.16	29.98	5.01	0.80
$\text{BERT}_{\text{BASE}}$	5.40	2.28	30.06	5.25	0.71
+ ITPT	4.37	1.92	29.42	4.80	0.68
$\text{BERT}_{\text{LARGE}}$	4.86	2.04	29.25	4.86	0.62
+ ITPT	**4.21**	**1.81**	**28.62**	**4.66**	**0.61**

Conclusion: BERT large achieves state of the art performance

Take away points

1. Document length problem can be overcome.

2. Use a decay factor for layer learning rates.

3. BERT produces state of the art results in classification.

4. Pre-train before fine-tuning.

Knowledge distillation

- Problem: BERT models are computationally expensive. Can the knowledge learnt be transferred to a simpler model?
- Knowledge distillation aims to achieve this.
- Train a model to minimize two terms:
  - Classification loss: binary cross entropy
  - Distillation loss: Kullback-Leibler divergence between class probabilities output by student and teacher models.
- The overall loss function for distillation becomes:

$$L = L_{classification} + \lambda \cdot L_{distill}$$

Distilled LSTM vs BERT: performance

#	Model	Reuters		AAPD		IMDB		Yelp '14	
		Val. F_1	Test F_1	Val. F_1	Test F_1	Val. Acc.	Test Acc.	Val. Acc.	Test Acc.
9	LSTM_{reg}	89.1 ±0.8	87.0 ±0.5	73.1 ±0.4	70.5 ±0.5	53.4 ±0.2	52.8 ±0.3	69.0 ±0.1	68.7 ±0.1
10	BERT_{base}	90.5	89.0	75.3	73.4	54.4	54.2	72.1	72.0
11	BERT_{large}	**92.3**	**90.7**	**76.6**	**75.2**	**56.0**	**55.6**	**72.6**	**72.5**
12	KD-LSTM_{reg}	91.0 ±0.2	88.9 ±0.2	75.4 ±0.2	72.9 ±0.3	54.5 ±0.1	53.7 ±0.3	69.7 ±0.1	69.4 ±0.1

Distilled LSTM vs BERT: inference time

Dataset	LSTM_{reg}	BERT_{base}
Reuters	0.5 (1×)	30.3 (60×)
AAPD	0.3 (1×)	15.8 (50×)
IMDB	6.8 (1×)	243.6 (40×)
Yelp'14	20.6 (1×)	1829.9 (90×)

Take away points

1. Document length problem can be overcome.
2. Use a decay factor for layer learning rates.
3. BERT produces state of the art results in classification.
4. Pre-train before fine-tuning.
5. BERT is computationally expensive for training and inference.
6. Knowledge distillation can reduce inference computational complexity at a small performance cost.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean.2015. Distilling the knowledge in a neural network. arxiv/1503.02531

Ashutosh Adhikari, Achyudh Ram, Raphael Tang, and Jimmy Lin. 2019. DocBERT: BERT for Document Classification. arxiv/1904.08398

Chi Sun, Xipeng Qiu, Yige Xu, Xuanjing Huang. How to Fine-Tune BERT for Text Classification? arxiv/1905.05583