Out of Order: How important is the sequential order of words in a sentence in Natural Language Understanding tasks?

Thang M. Pham[1] tmp0038@auburn.edu

Trung Bui[2] bui@adobe.com

Long Mai[2] malong@adobe.com

Anh Nguyen[1] anh.ng8@gmail.com

[1]Auburn University [2]Adobe Research

GLAMI AI

5.1.2021

Radek Bartyzal

Motivation

- BERT models surpassed humans on GLUE benchmark
- do they really understand language better?

Previous work

- pretrained BERT captures word-order information in the first three layers
- unknown whether BERT-based classifiers actually use word order information when performing NLU tasks
- incorporating additional word-ordering and sentence-ordering objectives into BERT training (StructBERT) could lead improvement on GLUE. However, StructBERT findings are inconclusive across different GLUE tasks and models.

Main contribution

- Are state-of-the-art BERT-based models using word order information when solving NLU tasks?
- If not, what cues do they rely on?

How:

- select classification tasks from GLUE
- train models on them
- give them shuffled sentences and see what happens

GLUE tasks

Corpus	|Train|	|Test|	Task	Metrics	Domain
			Single-Sentence Tasks		
CoLA	8.5k	**1k**	acceptability	Matthews corr.	misc.
SST-2	67k	1.8k	sentiment	acc.	movie reviews
			Similarity and Paraphrase Tasks		
MRPC	3.7k	1.7k	paraphrase	acc./F1	news
STS-B	7k	1.4k	sentence similarity	Pearson/Spearman corr.	misc.
QQP	364k	**391k**	paraphrase	acc./F1	social QA questions
			Inference Tasks		
MNLI	393k	**20k**	NLI	matched acc./mismatched acc.	misc.
QNLI	105k	5.4k	QA/NLI	acc.	Wikipedia
RTE	2.5k	3k	NLI	acc.	news, Wikipedia
WNLI	634	**146**	coreference/NLI	acc.	fiction books

Table 1: Task descriptions and statistics. All tasks are single sentence or sentence pair classification, except STS-B, which is a regression task. MNLI has three classes; all other classification tasks have two. Test sets shown in bold use labels that have never been made public in any form.

Used models

Pretrained models:

- BERT
- RoBERTa
- ALBERT
- all downloaded from Huggingface
- all 12-layers + 12 attention heads
- all pretrained on English

Finetuning for the classification tasks:

- add 1 layer on top of encoder and finetune whole model

Creating shuffled examples

- always shuffle only 1 sentence from the input (that can be 1 or 2 sentences)
- 3 ways of shuffling: shuffle ngrams for n = {1, 2, 3}
- make sure that the shuffled sentence is different from the original

Question from QQP dataset and its 3 modified versions (Q3 to Q1) created by randomly shuffling 3-grams, 2-grams, and 1-grams:

How can smoking marijuana give you lung cancer?

Q_3 lung cancer | marijuana give you | How can smoking ?

Q_2 smoking marijuana | lung cancer | give you | How can ?

Q_1 marijuana | can | cancer | How | you | smoking | give | lung ?

Creating experiment sets: dev-r and dev-s

- subset of the validation set = **dev-r**
- step 1: choose only sentences longer than 3 words
- step 2: only selected the examples that were correctly classified by the classifier.
- step 3: balanced the numbers of “positive” and “negative” examples by randomly removing examples from the larger-sized class

- **dev-s** is then created from **dev-r** by shuffling the first sentence

Sizes of experiment sets

Task Name	Task Type	Label	GLUE dev-set processing				dev-r
			(a) Dev set	(b) Step 1	(c) Step 2	(d) Step 3	Total
CoLA	Acceptability	"unacceptable"	322	287	154	154	308
		"acceptable"	721	675	638	154	
RTE	NLI	"not entailment"	131	131	72	72	144
		"entailment"	146	145	127	72	
QQP	Paraphrase	"not duplicate"	25,545	22,907	20,943	12,683	25,366
		"duplicate"	14,885	14,000	12,683	12,683	
MRPC	Paraphrase	"not equivalent"	129	129	101	101	202
		"equivalent"	279	279	255	101	
SST-2	Sentiment	"negative"	428	427	402	402	804
		"positive"	444	443	420	402	
QNLI	NLI	"not entailment"	2,761	2,741	2,500	2,500	5,000
		"entailment"	2,702	2,690	2,527	2,500	
STS-B	Similarity	N/A	1,500	1,498	N/A	N/A	1,498

Chosen GLUE tasks = all binary classification

Single sentence:

- CoLa = is this sentence grammatically correct?
- SST-2 = classify sentiment

Pair sentences:

- MRPC, QQP = are these 2 sentences semantically equivalent?
- QNLI = is answer to sentence 1 (S1) present in sentence 2 (S2)?
- RTE = textual entailment = can meaning of S2 be inferred from S1?

Experiment: How much is word order information required for solving GLUE tasks? (not much)

- Word-Order Sensitivity score (WOS): $s = (100 - p)/50$
- reported numbers are an average over:
  - 3 models per each task
  - 3 types of shuffling = per ngram
  - 10 random shuffles per shuffling type

- Result: except for grammar correctness - not much
  - consistent over all 3 models
  - 2-grams and 3-grams shuffling is basically ignored = not needed for GLUE?

Task	(a) Perf. on dev-r		(b) Perf. on dev-s			(c) Word-Order Sensitivity			(d) StructBERT improvements		
	Models	Baseline	1-gram	2-gram	3-gram	1-gram	2-gram	3-gram	$\text{BERT}_{\text{Base}}$	$\text{BERT}_{\text{Large}}$	RoBERTa
CoLA	100 (0.93)	50	50.69 (0.95)	53.98 (0.94)	56.36 (0.92)	**0.99**	**0.92**	**0.87**	**+4.9**	+4.8	+1.4
RTE	100 (0.81)	50	75.69 (0.80)	81.89 (0.80)	85.18 (0.79)	0.49	0.36	0.3	N/A	**+13**	-0.9
QQP	100 (0.96)	50	83.19 (0.94)	88.02 (0.94)	89.04 (0.95)	0.34	0.24	0.22	+0.7	+1.2	+0.5
MRPC	100 (0.97)	50	83.89 (0.94)	87.1 (0.95)	89.38 (0.95)	0.32	0.26	0.21	N/A	+3.9	**+1.7**
SST-2	100 (0.95)	50	84.04 (0.94)	88.35 (0.94)	90.56 (0.94)	0.32	0.23	0.19	+0.2	+0.3	+0.4
QNLI	100 (0.98)	50	89.42 (0.96)	93.85 (0.97)	95.32 (0.98)	0.21	0.12	0.09	N/A	+3	+0.3
STS-B	89.67	N/A	87.80	88.66	88.95	N/A	N/A	N/A	N/A	N/A	N/A

How confident are models when classifying shuffled examples? (a lot)

- only slightly less confident after shuffling
- for all tasks except grammar correctness

How sensitive are models trained to predict the similarity of two sentences? (not much)

- predicting similar meaning of 2 sentences should require knowing the sentence meaning = understanding it

Result:

- 83% of QQP models' correct predictions remained correct after shuffling
- => despite being trained explicitly on predicting semantic similarity of sentence pairs, models still exhibit naive understanding of sentence meanings

How important are words to classification after their context is shuffled (still a lot)

- BERT’s word embeddings are known to be highly contextual

Result:

- except for CoLA and RTE models, the contribution of individual words to classification is almost unchanged even after the context of each word is randomly shuffled

If not word order, then what do models rely on to make correct predictions? (nobody knows DL is magic)

- analyzed on tasks with lowest word-order sensitivity:
  - QNLI = question answering with question shuffled
  - STS-2 = sentiment classification on a single sentence (shuffled)

Hypothesis:

- model fixates on certain words / n-grams

STS-2

- at least 60% of the SST-2 dev-set examples can be correctly predicted from only a single top-1 salient word

S	the film 's performances are thrilling .	1.00
S_1	the film thrilling performances are 's .	1.00
S_2	's thrilling film are performances the .	1.00
S_3	's thrilling are the performances film .	1.00
S_4	's the film performances are thrilling .	1.00
S_5	performances are 's film thrilling the .	1.00

Figure 5: An original SST-2 dev-set example (S) and its five shuffled versions (S_1 to S_5) were all correctly labeled "positive" by a RoBERTa-based classifier with certainty confidence scores (right column).

- Why didn't they check sentences that cannot be classified by the top-1 word?
- **thrilling** is that top-1 salient word

QNLI

- for ~58% (i.e. 1,453 / 2,500) of QNLI “positive” examples: (1) there exist ≥ 3 words in the question that can be found in the accompanying answer
- and there are only **3 self-attention matrices** (out of 144 = 12x12) that capture these duplicate words between question and answer
- authors focus a lot on these 3 self-attention matrices:
  - QNLI models used self-attention to capture word-wise similarity in a (question, answer) pair to make decisions, regardless of how words are arranged
  - removing these matrices = drop accuracy significantly = no surprise
  -
- I say: meh
  - this doesn’t tell us the interesting stuff = how the BERT increases its accuracy beyond finding the duplicate words - that can be done easily by baselines - but those do not have that high final accuracy

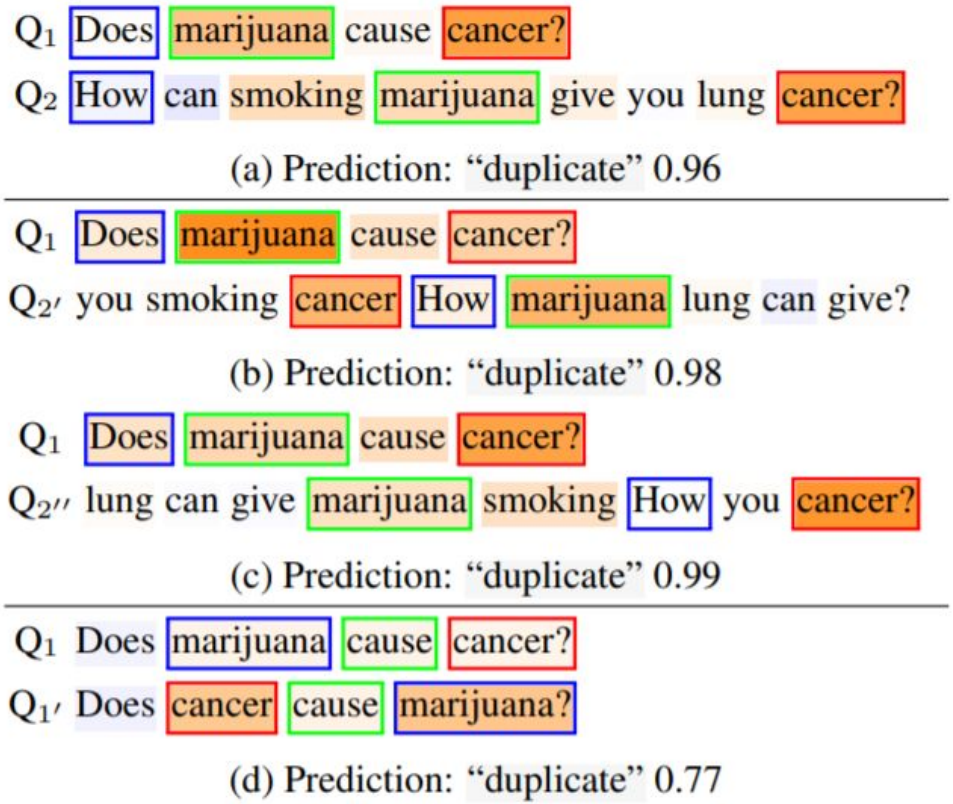
Does increasing word-order sensitivity lead to higher model performance? (yes slightly)

- first finetune the pretrained RoBERTa on a synthetic task
  - task = classify sentence to real / shuffled
- then re-initialize classification layer and finetune the model on a downstream tasks

- After the second finetuning on downstream tasks, all models were substantially more sensitive to word order
- results on downstrem tasks slightly improved
  - but that could have been caused by anything not just the word sensitivity

Example: RoBERTa on QQP

- labels semantically similar sentence same way even after shuffling it
- labels real sentence and its shuffled versions as duplicate



Conclusion

- words are more important than word order
- n-grams are even more important
- the models are capable of recognizing sentences out of order if explicitly trained to do so
- however they are not sensitive to it when trained on GLUE tasks
- => GLUE tasks are not that good at assessing this aspect of natural language understanding
- => with better models we need better benchmarks
- My 2 cents: we are getting close to needing a human in the loop
    - we are trying to imitate human understanding
    - humans do not have general intelligence
    - so we are trying to capture human knowledge+culture through NLP

Sources

- Main paper: https://arxiv.org/abs/2012.15180
- GLUE: https://arxiv.org/pdf/1804.07461.pdf
-