NLP Project Proposal

Andreas Kuster, Viswa Virinchi, Jakub Filipek

February 18, 2020

Citation

Title: ner and pos when nothing is capitalized

Url: www.aclweb.org/anthology/D19-1650/

Authors: Stephen Mayhew, Tatiana Tsygankova, Dan Roth

Abstract:
For those languages which use it, capitalization is an important signal for the fundamental NLP tasks of Named Entity Recognition (NER) and Part of Speech (POS) tagging. In fact, it is such a strong signal that model performance on these tasks drops sharply in common lowercased scenarios, such as noisy web text or machine translation outputs. In this work, we perform a systematic analysis of solutions to this problem, modifying only the casing of the train or test data using lowercasing and truecasing methods. While prior work and first impressions might suggest training a caseless model, or using a truecaser at test time, we show that the most effective strategy is a concatenation of cased and lowercased training data, producing a single model with high performance on both cased and uncased text. As shown in our experiments, this result holds across tasks and input representations. Finally, we show that our proposed solution gives an 8% F1 improvement in mention detection on noisy out-of-domain Twitter data.

1 Contributions

Exercise 1a

Task: A clear list of the scientific hypotheses evaluated in the original paper. Some papers don't make this super clear, so it can take a couple readings of the paper to understand.

The authors hypothesised that re-training standard NER and POS architectures with mixed cased and uncased training data would remove the dependence on

casing and improves the F-score of such classification tasks. They perform experiments on cased and uncased test sets, retraining the NER and POS models using different embedding by

- Training on cased data
- Training on uncased data
- Training on concatenation of cased and uncased
- Training on concatenation of cased and uncased but randomly lowercasing 50% of the data
- Training on cased and truecase the test data
- Truecase both the training and test data

They perform the same experiments with The Broad Twitter data, an informal corpus, retraining the NER model and verify their hypothesis that retraining on a mixture of cased and uncased datasets will improve F-scores

2 Access to data

Task: Give a short description of whether and how you can access the data used in the paper.

The paper describes different evaluations performed using several models and data sources. We will go through them, list all data sources, and give a summary at the end, where we describe how we can get hold of this data.

Evaluation 1

Performance evaluation of modern models trained on cased text, evaluated on lowercased text.

		Test	
Tool	Task	Cased	Uncased
BiLSTM-CRF w/ ELMo	NER	92.45	34.46
BiLSTM-CRF w/ ELMo	POS	97.85	88.66

Data sources:

- CoNLL 2003
- Penn TreeBank

Evaluation 2

Truecaser performance evaluation.

System	Test set	F1
(Susanto et al., 2016)	Wikipedia	93.19
BiLSTM	Wikipedia	93.01
	CoNLL Train	78.85
	CoNLL Test	77.35
	PTB 01-18	86.91
	PTB 22-24	86.22

Data sources:

- Simplified Wikipedia
- CoNLL 2003
- Penn TreeBank

Evaluation 3

NER+ELMo / POS+ELMo performance evaluation on CoNLL 2003 and PTB data.

Exp.	Test (C)	Test (U)	Avg
1. Cased	**92.45**	34.46	63.46
2. Uncased	89.32	89.32	89.32
3. C+U	91.67	89.31	**90.49**
3.5. Half Mixed	91.68	89.05	90.37
4. Truecase Test	82.93	82.93	82.93
5. Truecase All	90.25	**90.25**	90.25

Exp.	Test (C)	Test (U)	Avg
1. Cased	**97.85**	88.66	93.26
2. Uncased	97.45	**97.45**	97.45
3. C+U	97.79	97.35	97.57
3.5. Half Mixed	**97.85**	97.36	**97.61**
4. Truecase Test	95.21	95.21	95.21
5. Truecase All	97.38	97.38	97.38

Data sources:

- CoNLL 2003
- Penn TreeBank

Evaluation 4

NER+ELMo performance evaluation on twitter data.

Exp.	Mention Detection F1
1. Cased	58.63
2. Uncased	53.13
3. C+U	**66.14**
3.5. Half Mixed	64.69
4. Truecase Test	58.22
5. Truecase All	62.66

Data sources:

- Twitter

Conclusion

The data sources are: CoNLL 2003, Penn TreeBank, Simplified Wikipedia and Twitter. We were able to obtain a copy of the *CoNLL 2003* and *Penn TreeBank* dataset internally. Furthermore, the data from *Simplified Wikipedia* and *Twitter* is publicly available on the web. Therefore, we conclude that we obtained access to all required data for this project.

3 Feasibility

The project should be feasible because majority of models are either available pre-trained or with implemented architectures as authors specify exactly which implementation they used (mostly based on AllenNLP github repo). There will however be a few things such as Glove word vectors, which I could not find official implementation for (There is this unofficial attempt though, which would have to be checked for correctness). Hence, overall we should not be too heavy on GPU use, which is the biggest bottleneck in such project.

Considering the fact that BiLSTM models with pre-trained weights are less than 500mb, they should be easily able to fit in 8GB of memory of 1080TI. Since those are the biggest models used in paper, other should not be a problem either.

Another nice thing about this project is that every data-set used is cited, and available. One thing to note is that sometimes the labels may not be available (as in Wikipedia dataset example), but due to nature of casing tasks, labels can be created automatically at virtually zero cost.

Hence overall our work will be computationally feasible because it will mostly rely on fine-tuning models which is much cheaper than training them from scratch.

The only bottleneck can occur if there will be significant discrepancies between the paper and our results, specifically when using BiLSTM-CRF and ELMO. This can result in rather computationally heavy task of training these models from scratch. However, we do have a computational time buffer, which can be used in such case.