Hyperpartisan News Detection Using ELMo Sentence Representation Convolutional Network

Ye Jiang, Johann Petrak, Xingyi Song, Kalina Bontcheva, Diana Maynard

Outline

1. SemEval 2019 Task 4: Hyperpartisan News Detection.
2. Phase1: Early-Bird Submission.
3. Phase2: Final Submission.
4. Phase3: Recent works.

SemEval 2019 task 4

- Hyperpartisan News: expresses an extremely one-sided opinion or unreasoning allegiance to one party.
- Binary document level classification.
- **by-publisher** (automatic labelled, N=750K).
- **by-article**-train (manual labelled, N=645) and test (manual labelled, N=628, evaluation only.)

Phase 1: Early-bird submission

- Initial tryout: Train a model on the by-publisher, and test it on by-article.
- .Train_test_split(by-publisher): training set (600K), validation set (150K), test set (645 by-article).
- Model selection: light weight shallow CNN, RNN(LSTM+Attention).
- Embedding: GloVe (6 billion words, 300 dimensions).
- Accuracy on test set: CNN (59.41%), RNN (61.39%).

Phase 1: Early-bird submission

- Summary of initial tryout:

1. The model trained on by-publisher **might** not be helpful when it tested on by-article.

2. The performance of CNN is similar to RNN model, but training speed is much faster.

Phase 1: Early-bird submission

- Second tryout: Train a model on the by-publisher/by-article, test it on by-publisher/by-article.
- .Train_test_split(by-publisher): training set (600K), validation set (100K), test set (50K).
- .Kfold(by-article): 10-fold cross validation.
- Accuracy on by-publisher test set (50K): 70.64%.
- Accuracy on by-article (averaging accuracies from10-fold): 79.53%.

Phase 1: Early-bird submission

- Summary of second tryout:

1. Accuracies on leaderboard: 76.59% (by-article test), 64.35% (by-publisher test)
2. The model can be improved by training/testing on by-publisher or by-article separately.

- Can the model be improved by padding more initial tokens (200, 400, 1000, etc..)?
- Can the contextual word embeddings could improve the model?

Phase 2: Final submission

- Third tryout: Padding more initial tokens from documents.
- .Kfold(by-article): 10-fold cross validation.
- Accuracy on by-article (averaging accuracies from10-fold): 78.29% (200 tokens), 79.53% (400 tokens), 80.93% (800 tokens), OOM (1000 tokens).

Phase 2: Final submission

- Fourth tryout: Use ELMo generate sequence input, train the model on by-publisher and test it on by-article.
- Embeddings: ELMo (Original 2x4096_512_2048cnn)
- .Train_test_split(by-publisher): training set (80K), validation set (20K),test set (645 by-article).
- Accuracy on by-article: 56.43%

Phase 2: Final submission

- Fifth tryout: Fine-tune the fourth tryout model on by-article set.
- Embeddings: ELMo (Original 2x4096_512_2048cnn)
- .Kfold(by-article): 10-fold cross validation.
- Accuracy on by-article (averaging accuracies from10-fold): 81.89%

Phase 2: Final submission

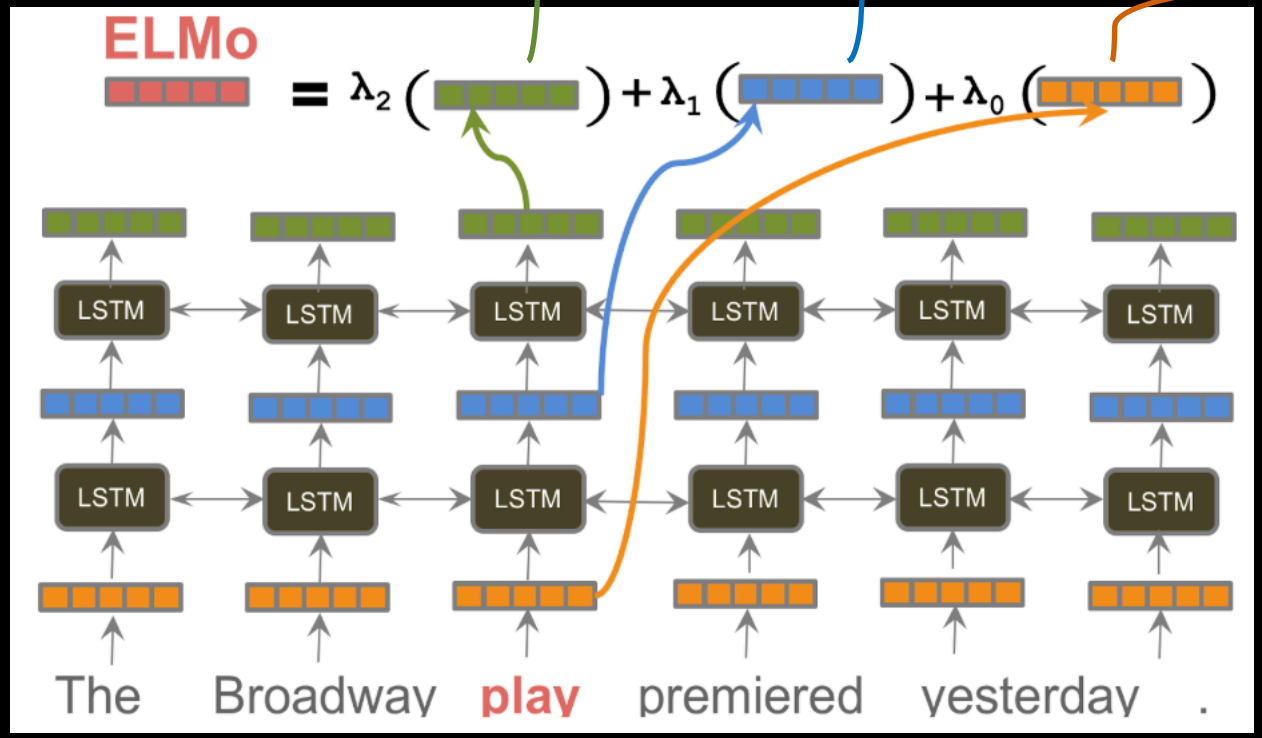
- Sixth tryout: Training on only the by-article and evaluate it.
- Embeddings: ELMo (Original 2x4096_512_2048cnn)
- Accuracy on by-article (averaging accuracies from10-fold): 83.87%

Phase 2: Final submission

- Summary of the fourth, the fifth and the sixth tryouts:

1. This confirms results that any use of the by-publisher data only hurts the model.
2. Contextual word embeddings improves model accuracy.
3. Boosting the accuracy to 84.04% by forming an averaged ensemble model.

Model structure



(peters et al, NAACL 2018)

[3, num_tokens per sentence, num _dimension]

Average

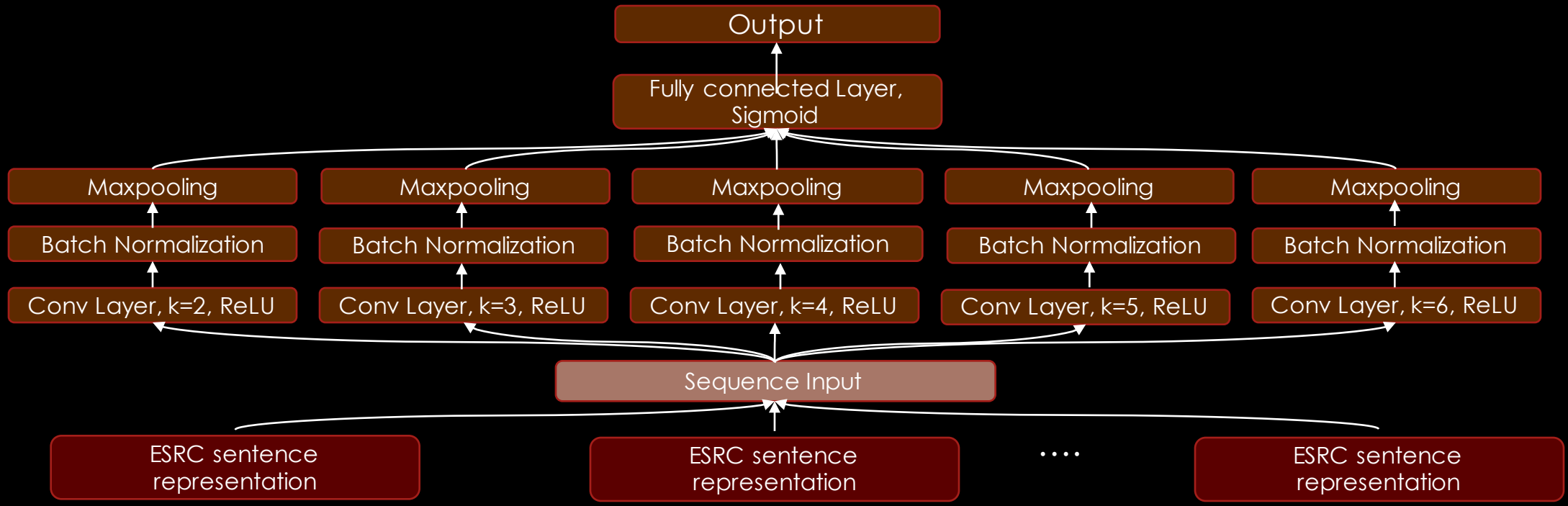
[num_tokens per sentence, num _dimension]

Average

[num _dimension]

ESRC sentence representation

Model structure



Phase 3: Recent works

1. Will BERT improve model accuracy again?

2. Can we use other approaches to train sentence representation instead of just taking the average of word embeddings?

3. Combine meta-data (LDA, word counts, sentiments, etc.) with sentence representation could also improve model accuracy.

THANKS

Questions?

- Email: yjiang18@sheffield.ac.uk

- Code: https://github.com/GateNLP/semeval2019-hyperpartisan-bertha-von-suttner
- Homepage: ye-jiang.github.io