Detecting Climate Change Misinformation with CRNN

Nodens F. Koren
University of Melbourne School of Computing and Information Systems
nkoren@student.unimelb.edu.au
Student ID: 1060811

Abstract

The impacts of climate change have become more prominent over the past years. Being able to identify the validness of climate articles hence became an important task. In this paper, we present our approach which combines a simple keyword filter with a deep learning model. According to the CodaLab competition ranking, our system came first out of 282 competitors. For misinformation detection, we attempted to combine the advantages of both CNN and RNN in order to extract high level features and model temporal dependencies.

1 Introduction

Climate change is a popular topic that started drawing attention over the past years. The number of new articles coming up each day drastically increased, but the validity of these articles is usually not examined. It is important to ensure the audience receive correct information; nonetheless, verifying the authenticity of climate change articles is a tough task. Without the knowledge from domain experts, it can be non-intuitive to tell if an article contains misinformation. In this paper, we propose a deep learning based system for detection of climate misinformation.

Recurrent Neural Networks (RNNs) are particularly useful when a sequence of data is being processed to make a classification decision or regression estimate (NVIDIA). They have been widely adapted into language related tasks such as language modeling, speech recognition, and machine translation (Keren and Schuller, 2016). They have also been showing promising results on detecting fake news (Bahad et al., 2019; Roy et al., 2018). However, it is a well-known issue that RNNs are difficult to train.

Convolutional Neural Networks (CNNs), on the other hand, are much easier to train when GPU optimization is available. The convolution mechanism works by extracting features from small local patches of the data and often is followed by a pooling mechanism to pool values of features over neighboring patches. CNN have yielded state-of-the-art results in the field of computer vision, but recently they have also been showing competitive results on NLP tasks such as speech recognition and sentence classification (Keren and Schuller, 2016). They have also been used to detect fake news (Yang et al., 2018). However, CNN alone is not strong at modeling temporal dependencies.

Previous works have attempted to combine the strength of both CNN and RNN so as to exploit both the high level features extracted by CNN and the temporal dependencies modeled by RNN. The combined model, CRNN, have been showing excellent results on numerous tasks including hyperspectral data classification (Wu and Prasad, 2017), music classification (Choi et al., 2017), and hand pose estimation (Hu et al., 2019). Keren and Schuller proposed a variant of CRNN which can take in sequences of varied length, and Roy et al. proposed a fake news detection system combining CNN and RNN based on ensemble methods. In our work, we attempted to combine the strength of both CNN and RNN serially. We adopted the original CRNN design which takes in sequences of fixed length instead of the variant proposed by Keren and Schuller.

Aside from the 1,168 default training data, an additional training set with 2,806 extra data was used. Reliable sources include The Guardian's Climate Change Section, NASA Global Climate Change, CNN News, BBC News, Skeptical Science, and DeSmog. Unreliable sources include Fox's News, National Review, The BFD, Skeptical Science's Climate Misinformers section, and DeSmog's Global Warming Disinformation Database. In addition to these, we also included approximately 200 satirical articles from Reddit, The Chaser, The Betoota Advocate, and The Shovel, which are some biggest satirical news distributors in Australia. See

Section 5.3 for detailed analysis.

According to the CodaLab competition ranking, our model came first out of 282 competitors.

2 Background

2.1. GloVe: GloVe is an unsupervised learning algorithm for obtaining vector representations for words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space (Pennington et al., 2014; Stanford NLP). There are several variants of GloVe, depending on the source of the corpus used in the training process, the number of tokens used, and the number of dimensions used to encode each word.

The GloVe model has been used by various NLP models yielding state-of-the-art results.

2.2. Attention Mechanism: While LSTMs possess the ability to learn temporal dependencies in sequences, they tend to be forgetful when dealing with long sequences and often fails to model dependencies in long sequences (Paradis and Whitmeyer, 2019). The attention mechanism can help the LSTM RNN learn these dependencies.

Attention is most commonly used in sequence-to-sequence models to attend to encoder states, but it can also be used in classification tasks. The attention mechanism works by calculating context vector c_i for each time frame based on the weighted sum of the previous hidden states $s_1, ..., s_m$: $c_i = \sum_j a_{ij} s_j$, where the attention score $\mathbf{a}_i$ is calculate based on the softmax of the selected variant of the attention function: $\mathbf{a}_i = softmax(f_{att}(\mathbf{h}_i, \mathbf{s}_j))$. The attention function calculates an unnormalized alignment score between the current hidden state $\mathbf{h}_i$ and the previous hidden state $\mathbf{s}_j$ (Ruder, 2017).

The variant we picked is the multiplicative attention, which is a simplified version of the original additive attention by calculating the following attention function: $f_{att}(h_i, s_j) = h_i^T \mathbf{W}_a s_j$, where $\mathbf{W}_a$ is a learned attention parameter (Ruder, 2017).

3 Our System for Misinformation Detection

In this section, we describe the system that we developed for climate misinformation detection.

3.1. Candidate Filtering: An article can be either climate-related or not. This can be considered

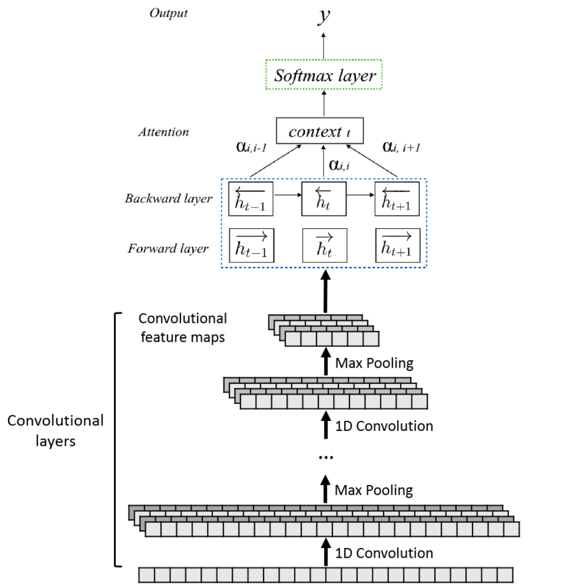

Figure 1: Our CRNN Model with an Attention Layer (Wu and Prasad, 2017; Schoene and Dethlefs, 2018)

an independent problem aside from climate misinformation detection as two sets of articles are disjoint. An article that does not relate to climate must not contain climate misinformation. The original tasks can thus be decomposed into two separate tasks: 1. classifying whether an article is related to climate, and 2. classifying whether a climate article contains misinformation.

We manually selected features by performing a Top K Frequent Words search on our training data and selected only climate-related keywords. We also exploited domain knowledge and added several unpopular words to our feature set. Our hypothesis behind is that doing so will alleviate our main model's duty and allows the main model to have higher fault tolerance on predicting irrelevant articles; the main model can thus focus on the classification of climate-related news.

3.2. Trainable Word Embedding Layer: We selected the uncased version of GloVe as our word embedding. As explained in Section 2.1, the word embedding layer links each word in the input article to a pretrained vector. We turned the trainable option on with the belief that this is more beneficial to domain-specific tasks.

The inputs were padded to a fixed size of 1,200 as this is the median length of the tokenized training sequences. We made this choice instead of using the max length (approximately 6,000) because the latter option will fill shorter sequences with padding characters, which can accidentally create undesirable similarity among them.

3.3. Convolutional Feature Map Extraction: We passed the embedding vectors into a deep convolutional subnetwork composed of three convolutional layers, each followed by a maximum pooling layer. The first layer contains 128 size-5 kernels, the second layer contains 128 size-4 kernels, and the third layer contains 128 size-3 kernels. The decision behind these hyperparameters is based on popular choices. The goal of this subnetwork is to build up high-level features that can be further used by the Bi-LSTM to do temporal analysis.

3.4. Bi-LSTM Temporal Analysis: We used 32 bidirectional-LSTM units to model temporal dependencies between feature maps. We believe feature maps will give the model a more general understanding of the context comparing to raw texts. These feature maps can be viewed as some kind of summaries and the recurrent layer's job is to analyze the temporal dependencies between these summaries.

The bi-directional mechanism was employed to capture complex long-term dependencies between the past and the future context. This is particularly important as articles can be written in numerous ways. Bidirectional-LSTM captures more possibilities by not only examining how the future context depends on the past, but vice versa.

A dropout rate of 0.25 and a recurrent dropout rate of 0.25 were used to prevent overfitting.

3.5. Multiplicative Attention Layer: Finally an attention layer was implemented to model global dependencies. As mentioned in Section 2.2, LSTM tends to be forgetful on very long sequence when doing classification task, and the Attention mechanism can mitigate this problem. The motivation behind is intuitive. In the English language, certain paragraphs serve more important roles than others do. For instance, in a typical five paragraph article, the introduction and the conclusion usually convey more information. It is thus reasonable to weigh context from these time-frames more than the rest.

The multiplicative variant of the Attention mechanism was utilized instead of the original additive version for efficiency reason. Multiplicative attention is faster and more space-efficient in practice. In our system, the Resource Exausted Error is inevitable when using additive attention.

3.6. Final Output: The final result is constructed by performing and AND operation on the feedback of the deep learning model and the candidate filter produced in Section 3.1 to ensure that the positive results predicted by the deep learning model are indeed climate-related.

4 Results

Table 1 shows the performance of our system on the given test and development sets in comparison to a 3-layer CNN, a 3-layer bidirectional LSTM, our proposed CRNN with a global max pooling layer in place of the attention layer, and our deep learning system without the candidate filter. Both the average F1 score and the best F1 score are reported for the test set performance.

Table 1: Development set performance of various models.

Model	F1	Precision	Recall
CNN	0.91	0.89	0.94
Bi-LSTM	0.88	0.80	0.98
GMP CRNN	0.92	0.88	0.96
A-CRNN	0.94	0.88	**1.00**
A-CRNN w/ Filter	**0.98**	**0.98**	0.98

The gradual improvement in the F1 score is a good evidence to support the idea behind our design. To verify that the improvement is not a sole result of more computational power, we further train deeper CNNs and Bi-LSTMs up to 10-layers. The F1 score did not improve with deeper CNNs or Bi-LSTMs, and this justifies our hypothesis that our CRNN indeed learned something new that cannot be achieved by CNN or Bi-LSTM alone. We also evaluate the effectiveness of the attention layer, and it turns out that the attention mechanism offers significant improvement in the F1 Score over the exact same model using a global max pooling layer instead, highlighting the importance of modeling global dependencies.

Furthermore we examine whether it is beneficial to make our deep learning model more specialized. Table 2 shows the performance of our system on the test set when trained under different settings. The model was trained separately on two different training sets. The first training set contains 50% climate articles and 50% non-climate articles, while the second training set contains 100% climate articles.

We can see that the specialized model works much better on detecting climate misinformation in comparison to the more general model. The 7% improvement in the precision is mostly from climate articles, as the candidate filter is static and identical under both settings.

Table 2: Test set performance of our system trained under different settings.

Dataset	F1	Precision	Recall
50%-Climate	0.82	0.70	**0.98**
100%-Climate	**0.88**	**0.79**	**0.98**

Finally we present the performance of our final model on the test set.

Table 3: Test set performance of our final model.

Leaderboard	F1	Precision	Recall
Public	0.93	0.88	1.00
Private	0.95	0.91	1.00

Surprisingly, our model performed even better on the private leaderboard. This is an indication that our system generalizes well enough to both sets of data and did not overfit to either the development set or the public leaderboard.

5 Error Analysis

In this section, we present the error analysis for each of the two sub-tasks, which can serve as a basis for further improvements of the system.

Due to the fact our system is already achieving 0.98 F1 score on the development set, we perform error analysis on an extra development set with 1,000 data, including posts from Skeptical Science and Reddit's climate change forum.

5.1. Missing Keyword in Filter: Even though our candidate filter is able to capture most of the keywords related to climate articles, there are always exceptions when facing unseen data. The main drawback behind our method is that the candidate filter detects only the words in the bag but not unseen words that have similar semantic meanings. This is the strength of word embedding and the filter could be replaced by another deep learning model for better generalization capability if the inclusion of more training data is permitted.

5.2. Prediction on Contradictory, Satirical and Ironic Languages: After further investigation, we found most of the errors made by our model fall into two specific categories.

The first category includes articles that frequently use contradictory structures. For instance, Skeptical Science tend to open their arguments with common climate denial claims, and counter them in the rest of the sections. Given that our system reads in only the first 1,200 words, it might not be able to accurately capture the big picture but instead only the opening sections containing climate misinformation. A possible solution to this problem is to augment our system in a way such that it can handle inputs of dynamic length.

The second category includes posts where satirical rhetoric and irony are extensively used, such as memes and ironic jokes. This is an error analysis we had done prior to the finalization of our model. We added about 200 satirical articles to our training data, whereas the model still predicts this type of articles incorrectly approximately 40% of the time. A pattern we observed is that our model predicts this type of articles with low confidence score. We consider this to be a hard problem even for human readers and associates it with the English rhetoric. An easy to understand example is, "Sea level will stop rising if there's no iceberg left. How good! At least those scooters don't block the sidewalks anymore." In spite of its literal meaning, the commenter was in fact trying to convey a message with a complete opposite meaning. This problem could potentially be relieved by focusing more on the punctuation marks or incorporating some confidence-based methods; however, we also found that adding more articles of this type has a negative effect on the recall score.

6 Conclusions

In this paper, we presented the system for climate misinformation detection. According to the CodaLab competition ranking, our model came first out of 282 competitors. The detection task was divided into two parts: (i) Determine if a given article is related to climate. (ii) Determine if a given climate article contains misinformation. We built a candidate filter by manually extracting features with the belief that doing so will alleviate our main model's duty and allow the main model to specialize on classifying climate-related news. We then implemented a CRNN, with the addition of the attention mechanism, and trained it using only climate-related data so as to make it more specialized. The combination of these two systems significantly outperforms the baseline.

References

Pritika Bahad, Preeti Saxena, and Raj Kamal. 2019. Fake news detection using bi-directional lstm-recurrent neural network. *Procedia Computer Science*, 165:74 – 82. 2nd International Conference on Recent Trends in Advanced Computing ICRTAC - DISRUP - TIV INNOVATION , 2019 November 11-12, 2019.

BBC News. https://www.bbc.com/news.

Keunwoo Choi, György Fazekas, Mark Sandler, and Kyunghyun Cho. 2017. Convolutional recurrent neural networks for music classification. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2392–2396. IEEE.

CNN. https://www.cnn.com/.

DeSmogBlog. https://www.desmogblog.com/.

EnvironmentalScienceDegree.com. 101 top web resources on climate change. https://www.environmentalsciencedegree.com/climate-change/.

Fox News. https://www.foxnews.com/.

George Washington University Public Health. 2020. 54 great sources for climate change news. https://publichealthonline.gwu.edu/blog/sources-for-climate-news/.

Zhongxu Hu, Youmin Hu, Jie Liu, Bo Wu, Dongmin Han, and Thomas Kurfess. 2019. A crnn module for hand pose estimation. *Neurocomputing*, 333:157 – 168.

Gil Keren and Björn Schuller. 2016. Convolutional rnn: an enhanced model for extracting features from sequential data. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 3412–3419. IEEE.

NASA: Climate Change and Global Warming. https://climate.nasa.gov/.

National Review. https://www.nationalreview.com/.

NVIDIA. https://developer.nvidia.com/discover/recurrent-neural-network.

Samuel Paradis and Michael Whitmeyer. 2019. Pay attention: Leveraging sequence models to predict the useful life of batteries. *ArXiv*, abs/1910.01347.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Reddit Inc. https://www.reddit.com/.

Arjun Roy, Kingshuk Basak, Asif Ekbal, and Pushpak Bhattacharyya. 2018. A deep ensemble framework for fake news detection and classification. *arXiv preprint arXiv:1811.04670*.

Sebastian Ruder. 2017. https://ruder.io/deep-learning-nlp-best-practices/.

Annika Marie Schoene and Nina Dethlefs. 2018. Unsupervised suicide note classification .

Skeptical Science. https://www.skepticalscience.com/.

Stanford NLP. https://nlp.stanford.edu/projects/glove/.

The Betoota Advocate. https://www.betootaadvocate.com/.

The BFD. https://thebfd.co.nz/.

The Chaser. https://chaser.com.au/.

The Guardian. https://www.theguardian.com/.

The Shovel. https://theshovel.com.au/.

Hao Wu and Saurabh Prasad. 2017. Convolutional recurrent neural networks forhyperspectral data classification. *Remote Sensing*, 9(3):298.

Yang Yang, Lei Zheng, Jiawei Zhang, Qingcai Cui, Zhoujun Li, and Philip S Yu. 2018. Ti-cnn: Convolutional neural networks for fake news detection. *arXiv preprint arXiv:1806.00749*.