Bias Detection in Sports Articles Using Structured Data

By Sherman Ng (SUNET ID: sqng629)

Email: sqng629@gmail.com, sqng629@stanford.edu

Link to video of the presentation on Youtube: https://youtu.be/qoCIY_VhJp8

Overview and Problem Statement

Bias detection in news articles is a very difficult task where much research is currently being done. There have been quite a few different attempts at creating language models using machine learning and deep learning techniques such as linear discriminant analysis, recurrent neural networks, and LSTM. This experiment attempts to provide a solution to bias detection not by using a language modelling approach, but rather a methodology focused around using a structured text input feature approach to classify whether or not a particular article is biased. The methodology being tested is centered around first extracting text features, such as the text complexity and the number of words of certain parts of speech using tools such as the Stanford POS tagger, and then passing the structured input into some deep learning network architecture to classify the article.

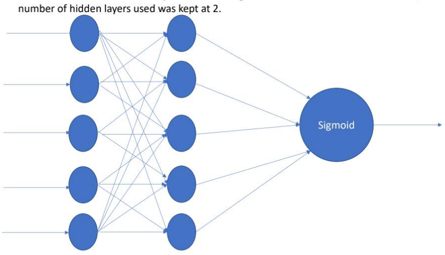
Dataset and experiment setup

This experiment specifically focuses on bias detection in sports news articles, and the dataset used is a database from the UCI Machine Learning Repository [1] containing 1000 parsed sports articles where part of speech and other text features such as the frequency of exclamation or question marks in the article, the frequency of pronoun use, and the number of foreign words used in the article are extracted. Each article in the database contains 59 total extracted features, so the full input dataset to the algorithms developed in this experiment can be viewed as a 1000 by 59 matrix. 95% of the dataset is used to train the network, while 5% of the remaining data is used as a validation dataset to measure the effectiveness of the trained network.

Article #	Frequency of plural proper nouns	Frequency of base form verbs	Frequency of genitive markers	Frequency of commas	Frequency of foreign words	Frequency of full stops	Text complexity score	…………
16	30	45	20	40	5	20	18	

Simple Neural Network

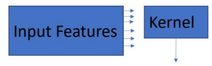
The first model that I experimented with is a simple neural network comprised by 2 fully connected hidden layers, non-linear activation functions at the output of the hidden layers, an output layer, and a sigmoid activation function at the output. An Adam optimizer was chosen for optimization of the weights of the neural network, and Xavier initialization was used for initialization of the weights and zero initialization for the biases. Initially, I used ReLU activation functions in the output of the hidden layers. Eventually, it was determined that using sigmoid activation functions at the outputs of hidden layers instead produced a better validation accuracy rate. Adding more than 2 hidden layers to the network did not seem to improve validation classification accuracy while increasing the time it takes to train the network, so the number of hidden layers used was kept at 2.



Hidden layers with sigmoid activation

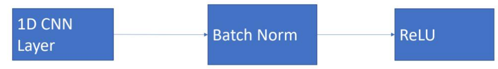
Convolutional Neural Network Architecture

Another model that I experimented with in an attempt to create an architecture that potentially generalizes better to unseen examples is a multi-layer convolutional neural network. The 59 input features for each training example are passed as inputs into a network block where the first layer is a 1-dimensional convolutional layer, with kernel sizes and the number of filters determined empirically. In the first convolutional layer, for each training example, c number of m x 1 kernels are convolved with the input features using "same" padding to make sure that input features on the edges do not get dropped out (c and m refer to the number of filters in the convolutional layer and kernel size respectively). The rationale behind using an approach like this is based upon the hypothesis that certain input features may be correlated with each other. Using a CNN approach may enable the model to better capture the interdependencies between the features:

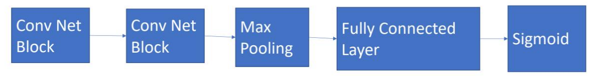


The output of this layer, with dimension p x c x 1 where p is the number of input features, is then passed to a batch normalization layer for regularization purposes.

Each convolutional network block consists of three basic components: a 1-dimensional convolutional layer, a batch normalization layer, and a non-linear activation layer:



Architecture of the full network:



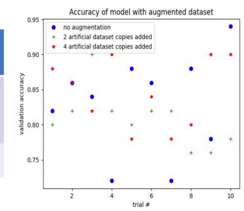
Results

Possibly because of the small number of training examples available for use in training the models, there is quite a bit of fluctuation in the validation dataset accuracy each time a model is trained and evaluated. To get a clearer picture of how a model performs when a certain set of hyperparameters is used, a model is trained 10 times on each set of hyperparameters used, and each time a model is trained, a random 5% of the dataset is selected as the validation set while the remaining 95% is used for training. The performance of the model under a set of hyperparameters is then noted over the 10 trials.

Since this experiment attempts to classify inputs into a binary output, a binary cross entropy loss function was used to evaluate the performance of both models:

$$\text{Loss} = -(y * log(p) + (1-y)\log(1-p))$$

	Training Accuracy	Validation Accuracy
Simple Neural Network	98%	83%
CNN	99%	84.6%



Simple Neural Network

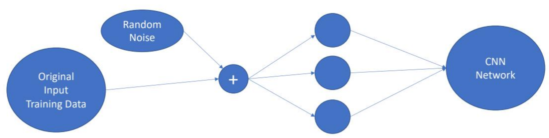
Hyperparameters tuned:
-number of layers in the network (2)
-dropout keep probability (0.6-0.9)
-Number of hidden nodes in each layer (300 in the 1st, 200 in the 2nd)

Convolutional Neural Network

Hyperparameters tuned
-Number of convolutional kernel filters (5)
-kernel size (3)
-Number of convolutional network blocks (2)

Regularization and Data Augmentation

Artificially created input feature = original feature + rand(-d, d)



Conclusion

From the results of this experiment, there does not appear to be clear cut evidence of the superiority of either the simple neural network architecture or the convolutional neural network architecture in determining bias in sports news articles. Optimizing hyperparameters for either model also did not significantly improve classification accuracy. However, from the data augmentation experiment, it appears that perhaps by either artificially generating or collecting more training data to train the models, the convolutional neural network architecture may be able to generalize and classify unseen articles better. This provides hope that given more time to either further tune the data augmentation techniques or gather and parse more training data, a more accurate deep learning based model that takes text features as input can be developed and trained with higher classification accuracy. The difficult problem of bias detection in news articles can then essentially be broken down into 2 steps, first by extracting text features from news articles and then by passing extracted features into the aforementioned model.

References:

[1]: UCI Machine Learning Repository: Flags Data Set, 9 Apr. 2018, archive.ics.uci.edu/ml/datasets/Sports+articles+for+objectivity+analysis.

[2] Recasens, M., Danescu-Niculescu-Mizil, C., & Jurafsky, D. (2013). *Linguistic Models for Analyzing and Detecting Biased Language*. Retrieved from Association for Computational Linguistics website: https://nlp.stanford.edu/pubs/neutrality.pdf

[3] DeVries, T., & Taylor, G. W. (2017). *Dataset Augmentation in Feature Space*. Retrieved from ICLR website: https://openreview.net/forum?id=HJ9rLLcxg

[4] Doumit, S., & Minai, A. (2011). *Online News Media Bias Analysis using an LDA-NLP Approach*. Retrieved from https://eecs.ceas.uc.edu/~aminai/papers/doumit_iccs2011.pdf

[5] Hamborg, F., Meuschke, N., & Gipp, B. (2018). *Bias-aware news analysis using matrix-based news aggregation*. Retrieved from https://www.researchgate.net/publication/325228442_Bias-aware_news_analysis_using_matrix-based_news_aggregation

[6] Ioffe, S., & Szegedy, C. (2015). *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. Retrieved from https://arxiv.org/pdf/1502.03167v3.pdf