# Fake News Detection in the Health Domain

Ojaswi Binnani 20161006

Abheet Sharma 20161091

## PURPOSE

Fake News is news articles that are spread in social media with catchy titles and exaggerated or misleading content. It relies on a person's confirmation bias - the phenomenon that happens unconsciously in a person's brain where news that fits with a preconceived notion is regarded as true regardless of the authenticity of the news article, whereas something that may be against the person's original information is considered as false even if the news article is backed with facts and research. It is easy to fall for news articles, and there are a plethora of fake-news articles out on the internet to prey on this fact.

## MOTIVATION

In the health domain, it is harder to fact-check and it is more important for the reader to get the correct information about their health. Health-related information tends to be more personal than other news domains such as celebrity gossip and requires technical expertise in confidently knowing correct information from wrong or exaggerated information.

Our aim was to develop a model that uses linguistic differences between real-news articles and fake-news articles to determine whether a given article is real or fake rather than using fact-checking programs so that the model can be used for articles about subjects that do not have much information about them.

## DATASET

For the purposes of our project, we used a FakeHealth Corpus which contains the original news articles, the review of each article scraped from HealthNewsReview.ord, the social engagement/discussions on Twitter about each article and reviews from experts in the medical field. There are about 2100 news articles that are present in the corpus. Each article is reviewed by experts on a scale of 1 to 5 based on ten different criteria.

We use the reviews of the experts to classify the corpus into fake news articles and real news articles by using a threshold review value to differentiate between both. We use two different threshold values: 2 and 3. We run our model on both threshold values and show the accuracies of the various models implemented.

# FEATURES

## EMOTION COGNIZANCE

Emotion cognizance is finding the emotion present in an article. The thought process behind this feature is that publishers of fake news tend to articulate their news such that it elicits powerful emotions in the reader (such as anger, disgust, sadness) in order to make the reader interested to continue reading the fake news article.

There is another advantage to this feature as well. It learns the base linguistic architecture of fake news which remains similar or constant as time varies, since most fake news tries to/will try to elicit powerful emotions to hook readers. Such an advantage is not present in other approaches such as the Knowledge Base approach, which requires the knowledge base to be continuously updated (expensive task).

We use NRC-emotion-intensity-lexicon where data is a three-tuple in the format (w, e, s) where w is the word, e is the emotion associated with the word, and s is the intensity of the emotion. If we take the original document to be D, we copy each word and its emotion (if the intensity ‘s’ of the emotion is higher than a certain threshold, in our case the threshold is 0.6) into a document D’. This new D’ is our new enlarged emotionized text-representations.

### DOC2VEC

Now that we have D (the original documents) and D’(the enlarged documents), we train two separate Distributed-Bag-Of-Words DOC2VEC models for each D and D’ to yield vectors V and V’ respectively.

### SENTIMENT ANALYSIS

Sentiment Analysis is giving the article a score on whether the article is positive, negative or neutral. The thought process behind this feature is that perhaps fake-news articles would be more extreme in its sentiment (very positive or very negative) whereas real-news articles would be more neutral. We train a model on positive and negative articles and then run our articles through that model to receive a positive or negative score for each article and add it to the vectors.

## METHOD

1. The first step was to prove that Emotion Cognizance is useful for our chosen dataset. So, we first obtain DBOW DOC2VEC features (explained in FEATURES section) on the original articles as V, and the new enlarged emotionized articles as V’. We then trained two SVMs with RBF kernel on V and V’ separately and evaluated the two models on the dataset. Since the ratings are between 0-5, we reduce it to 0,1 based on if the rating is above a threshold. We do two thresholds, t=2 and t=3. The results we achieved are

**SVM WITH EMOTION COGNIZANCE**

Threshold = 2: Accuracy: 91.3%

Threshold = 3: Accuracy: 62.43%

**SVM WITHOUT EMOTION COGNIZANCE**

Threshold = 2: Accuracy: 90.37%

Threshold = 3: Accuracy: 60.74%

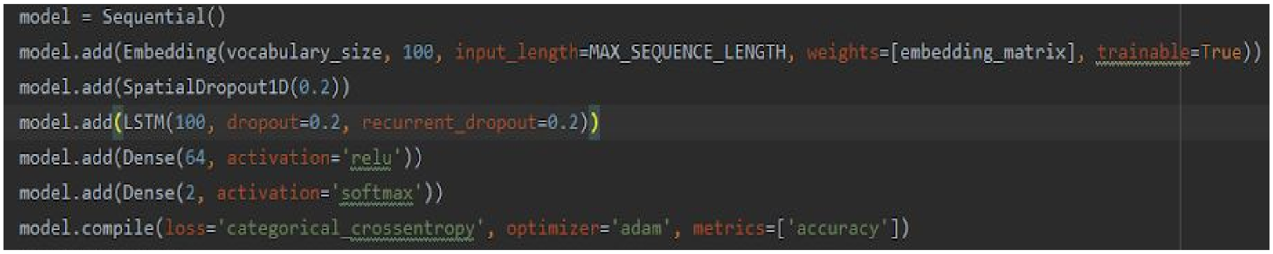
Here, we see that the SVM trained on emotionized text gave better results. Thus, we can assure that Emotion Cognizance is useful in our results.

2. After proving that Emotion Cognizance is useful for our dataset, we decided to build two separate LSTM models (one basic, and the other a Bi-LSTM model) both pre-trained on Glove weights. We then train our dataset on the enlarged emotionized text-representations split into train-test-val, and evaluate our models on test split and validation split. We noted the accuracies.
3. So far, we have added a Linguistic Approach to our model in the form of emotion cognizance. Our BiLSTM model is a faux-Machine Learning Approach(not deep enough, but still powerful). Finally, we thought of adding a Network/Human Approach to our model by utilizing Twitter engagements on each of the news articles. We apply the same pipeline as discussed earlier, but to each of the tweets, and add it as a parameter to our models. We noted the accuracies.

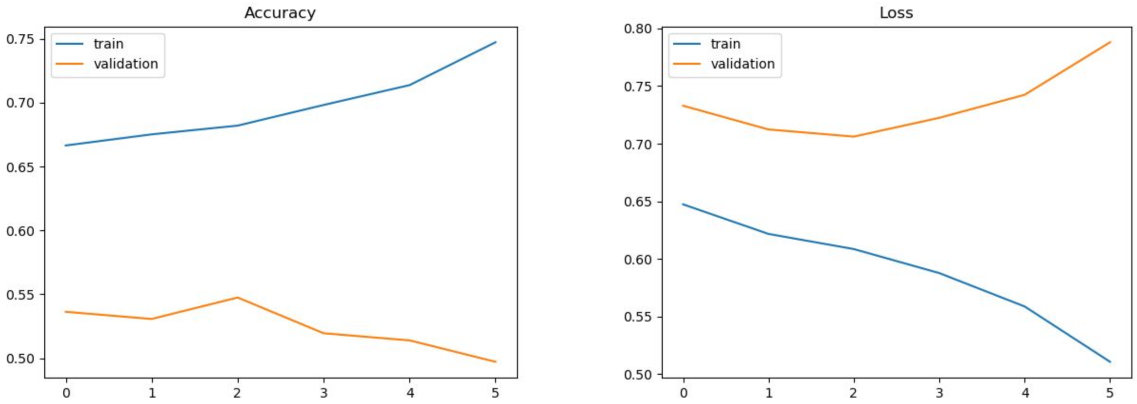
# MODELS

## MODEL 1

Base LSTM with dropout pretrained on Glove weights

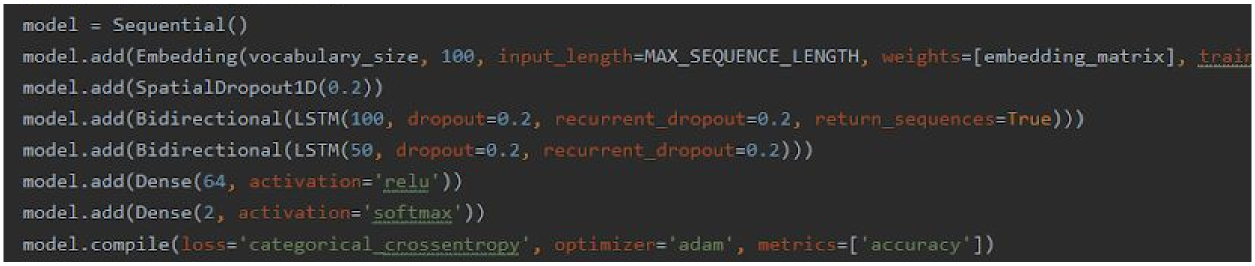


```
model = Sequential()
model.add(Embedding(vocabulary_size, 100, input_length=MAX_SEQUENCE_LENGTH, weights=[embedding_matrix], trainable=True))
model.add(SpatialDropout1D(0.2))
model.add(LSTM(100, dropout=0.2, recurrent_dropout=0.2))
model.add(Dense(64, activation='relu'))
model.add(Dense(2, activation='softmax'))
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
```

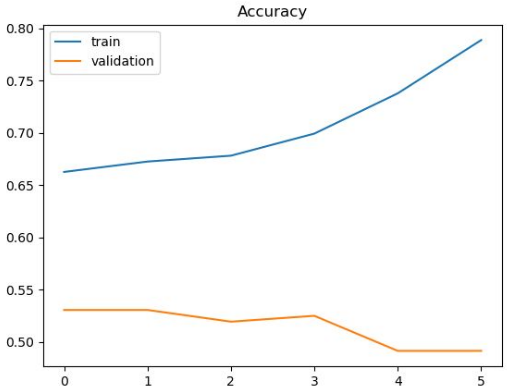


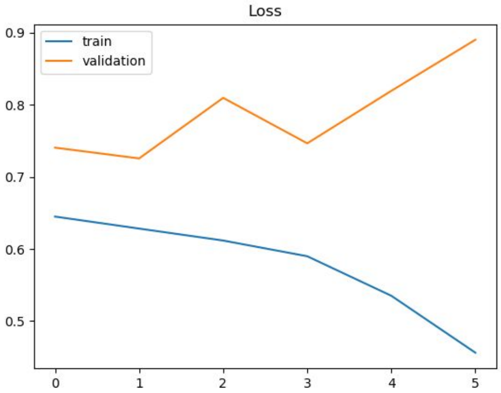
## MODEL 2

BiDirectional LSTM pretrained on Glove weights along with dropout pretrained on Glove weights was implemented to see if we can boost base accuracies.

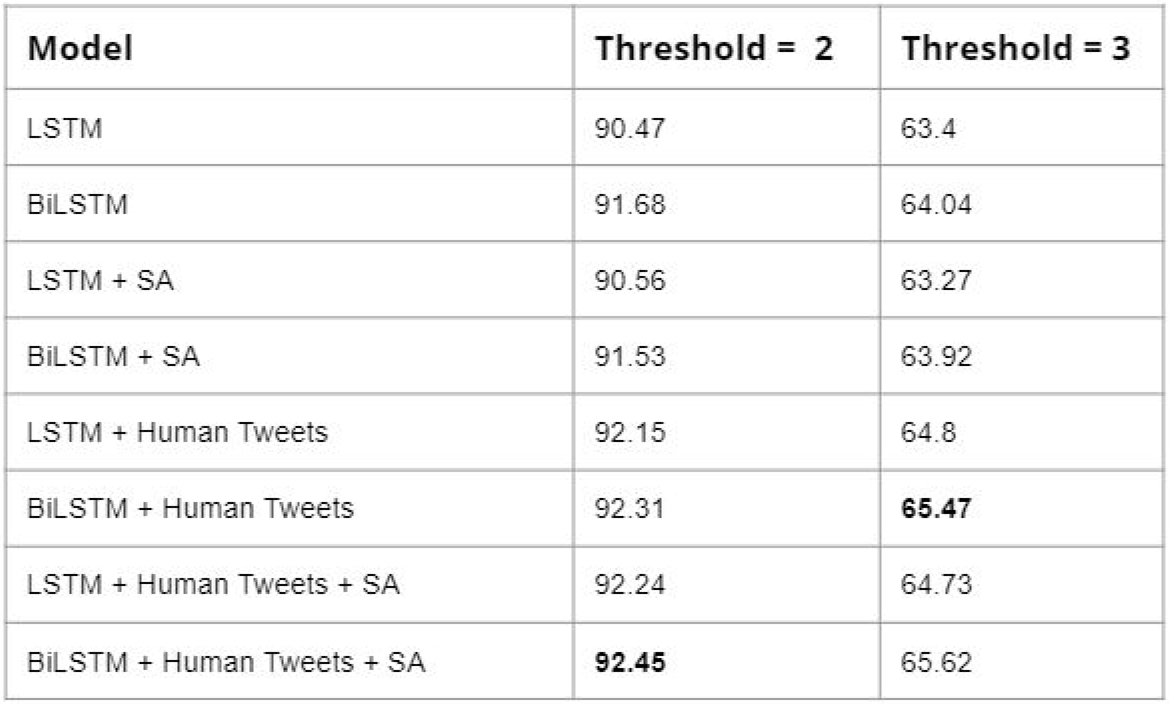


```
model = Sequential()
model.add(Embedding(vocabulary_size, 100, input_length=MAX_SEQUENCE_LENGTH, weights=[embedding_matrix], trai
model.add(SpatialDropout1D(0.2))
model.add(Bidirectional(LSTM(100, dropout=0.2, recurrent_dropout=0.2, return_sequences=True)))
model.add(Bidirectional(LSTM(50, dropout=0.2, recurrent_dropout=0.2)))
model.add(Dense(64, activation='relu'))
model.add(Dense(2, activation='softmax'))
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
```





# RESULTS



| Model | Threshold = 2 | Threshold = 3 |
|---|---|---|
| LSTM | 90.47 | 63.4 |
| BiLSTM | 91.68 | 64.04 |
| LSTM + SA | 90.56 | 63.27 |
| BiLSTM + SA | 91.53 | 63.92 |
| LSTM + Human Tweets | 92.15 | 64.8 |
| BiLSTM + Human Tweets | 92.31 | **65.47** |
| LSTM + Human Tweets + SA | 92.24 | 64.73 |
| BiLSTM + Human Tweets + SA | **92.45** | 65.62 |

# CONCLUSION

From our initial SVM Model, and then further creating a LSTM and Bi-LSTM model with different sets of features, we can conclude that emotion cognizance plays a key role in differentiating between real-news and fake-news articles.

Further, we see that sentiment analysis does not help in significantly increasing accuracies. We theorize that this happened because Sentiment Analysis is very basic emotion mining (+ve, -ve), and our emotion cognizance approach already covers this field quite well. We can see from the table above that adding sentiment analysis increases the accuracies by around 0.2 percent, and hence it is not significant or necessary.

Finally, our last step of adding emotion cognizance to twitter discussion/engagement as a feature to our model, we can see that the accuracies of our models with both threshold values increases and hence we suggest adding tweets around an article into the model as well.

## FURTHER WORK

There are other features we could use and test whether the additional features help the model in its prediction of other articles. We could use individual scores from each of the criteria in the training model and see if individual criteria are a more accurate way of creating a split between real-news and fake-news articles.

Each news article has its own review in our dataset. We could try to match the emotions in the full fledged review to that of the user reviews directly for better detection.

We could also utilize deep neural structures to fully take advantage of our dataset. The base idea “Emotion Cognizance” is widely applicable, and it is very easy to fit it in SotA models.

There are other metadata marked with each news article such as Links, references etc. which we could also add in our model.

Finally, the dataset also has the user networks of the tweet engagements. There exists an unsupervised method which utilizes user networks for fake news detection. We could also add this feature as well.