Group final project - BAN400 fall 2020

Candidate: (. . .)

6/11/2020

Contents

Project suggestion Group 30

Initial proposal:

"Textual data analysis combined with regression analysis using data on fake/true news"

Method:

- We create regressors based on the words present in the fake and real news. So far we can create a sentiment factor, on both the body and the title of the article, and a keyword per word factor, also both on the title and the body.
- Then use a predictive model to see if the news are real or fake based on these factors. We can run a factor regression or perhaps a machine learning model, like an XGBoost model.
- Finally, we can create a "shiny ap" which allows up to paste news articles in, the ap will then preprocess the article and give us a score of "fake probability".

Analysis:

- Tokenizing and prepossesing before doing any textual analysis, we need to preposess the data, which mean shaping it in order for the different models to read them.
- Topic modelling: Using the words in the dataset to define a concrete topic in each of the files. This could help us see which words to look out for, perhaps there is a topic which is "Hype up words", that might be more present in the fake news. We can create a "topic per word" score.
- Sentiment analysis to gather data on how negative /positive the fake news is compared with true news. Maybe there is a specific sentiment in the fake news that we can extract, we can add these numbers to our regression, to make a predictive model to see if the news are fake or not.
- Run a regression, were 1 is fake news and 0 is true news. It could be a linear regression, or we could experiment with some machine learning model. We keep whichever performs best.

Sources

For this we will use a data set from Kaggle: https://www.kaggle.com/clmentbisaillon/fake-and-real-news-dataset (Links to an external site.) with data from 2016 to 2017.

All graphs will be visualized using ggplot package.

- Use machine learning
- Apply it to live data

https://www.r-bloggers.com/2020/10/sentiment-analysis-in-r-with-custom-lexicon-dictionary-using-tidytext/"

https://rstudio-education.github.io/tidyverse-cookbook/import.html https://www.tidyverse.org/blog/2020/06/recipes-0-1-13/

Workflow:

1. get the data to run in R and tidyverse

1.0 merge true and fake news datasets, adding a new dummy column if news is true or fake.

1.1 separate the data into training and test data.

2. Prepossesing and getting the data ready for modelling.

—- Cleaning the data —-

Here we do some simple prepossesing steps, like removing characters and words that don't carry any meaning to our analysis.

—- Tokenizing the data and creating bigrans —-

Here we will split the data word by word. We will do the analysis with unigrams (which is every word separated individually), and bigrams (which are words separated in twos). They may give us slightly different topics, though should be somewhat similar, so its important to check for overfitting if using both in a regression.

—- Making a KWIC table —-

this helps us better visualize the text in hand before making a topic analysis

```
## # A tibble: 15 x 3
##    left                          keyword  right
##    <chr>                         <chr>    <chr>
##  1 truth charlottesville condemn leftist  terrorists blm antifa
##  2 carney august reaction        leftists criticism vogue cover
##  3 just know john                leftists aren opposed immigration
##  4 american version wasn         leftists made lady liberty
##  5 liberty anti right            leftist  bias completely politically
##  6 wish rambobiggs violent       leftist  caught taking shot
##  7 realdonaldtrump needs purge   leftists executive branch disloyal
##  8 king stevekingia march        leftists disloyal congressman king
##  9 groups violent strains        leftist  ideology even paid
## 10 marxist took life             leftist  guy now see
## 11 thought real gave             leftists cia cia usa
## 12 progressive reality today     leftist  movement made much
## 13 democrats stayed home         leftists refused unify can
## 14 anymore sounds like           leftist  whining sexism time
## 15 minh viet cong                leftist  secret squirrel stuff
```

```
## # A tibble: 15 x 3
##    left                           keyword         right
##    <chr>                          <chr>           <chr>
##  1 <NA>                           trump           just couldn wish
##  2 year america donald            trump           realdonaldtrump december trump
##  3 america donald trump           realdonaldtrump december trump tweet
##  4 trump realdonaldtrump december trump           tweet went welll
##  5 petty infantile gibberish      trump           lack decency won
##  6 pollitt korencarpenter decemb~ trump           new year eve
##  7 know love donald               trump           realdonaldtrump december noth~
##  8 love donald trump              realdonaldtrump december nothing new
##  9 december nothing new           trump           years trump directed
## 10 new trump years                trump           directed messages enemies
## 11 dale ddale december            trump           holiday tweets clearly
## 12 security secretary donald      trump           administration email scandal
## 13 message rebuke donald          trump           without even mentioning
## 14 united nations condemned       trump           move recognize jerusalem
## 15 forced homes issue             trump           continues fight francis
```

2.1 cleaning the words, and making them ready for the sensitivy analysis(making a frequency list)

2.1.1 Use recipes in order to make this cleaning process replicable for test data

2.2 Using the sensitivity analysis to make usable columns in a tibble for each news article.

2.3 gathering all the data into one tibble, which we'll use to train our machine learning model.

3. Training the machine learning model

4. Test the model, and polish it (maybe go back to step 2 if need be)

5. make a shiny app, which will work as an interface to plug in new data and see if the news stories is true or fake.

5.1 make the app more user friendly