

US Fed FOMC Meeting minutes - Applying NLP algorithms to Predict Fed Fund Rate Decision - Results Summary

Objective

Observe and improve the results of applying NLP algorithms on FOMC meeting minutes to predict Fed Fund rate decision (hike, no hike, cut).

Data and variations

- Fed FOMC meeting minutes for 16 years (1999-2018) were used as the tagged feature sets and fed fund rate decision (hike, no hike, cut) was the tagged classification criteria
 - Note that some hike years' minutes were not used (2010-2013) as the corpus of no hike words/ngrams was becoming too large without any additional benefit
- Natural Language Processing using the NLTK library was done on the minutes documents as they are primarily text.
- All words in the text documents were tokenized and lemmatized to remove inflectional endings and reduce them to their base or dictionary form.
- An enhanced nltk stopwords list (general and FOMC statements related) was used to remove useless words that are of no use in training the models
- The tokenized words in each document were tagged based on the Fed Fund rate decision for it (hike, no hike, cut).
- The tagged words were converted into a feature set based on a subset of the highest frequency from the frequency distribution of all words in the documents
- **Note** - It is important to note that random sampling (shuffling) of the documents was done as there are more no rate hike and rate hike documents compared to rate cut documents
 - A random sample of the excess documents equal to the number of rate cut documents was used
 - This ensures that we are comparing equal no of documents to create the tokenized words for the tagged feature set
 - The documents and tagged words must be shuffled randomly multiple times to improve accuracy
- All data was taken from the Federal Reserve's FOMC website -
<https://www.federalreserve.gov/monetarypolicy/fomccalendars.htm> and
https://www.federalreserve.gov/monetarypolicy/fomc_historical_year.htm

Algorithms and variations

- Standard NLTK, numpy libraries were used for running the different algorithms in python
- Two approaches were taken to determine the type of algorithms to apply to the FOMC minutes documents
 - Classification algorithms were applied by tagging the tokenized words
 - Ngram (trigram, fourgram, fivegram) matching (non vectorized) frequency distribution of tokenized words
- **For Classification algorithms Naive Bayes and Decision Trees models** were used
 - 40% of the tokenized words were used to build the feature set as that gave the best results, instead of using the entire set as that gets quite large

- The data was split into test and training sets, with test_size = 0.15, as that gave better results compared to higher values
- Models were trained and tested on the feature set
- Predicting the fed fund rate decision for a new minute's document (not included in the training/testing feature set) was done using the trained model
- **Decision Trees gave better results compared to Naive Bayes**
- **Accuracy between 0.6 to 0.8 can be achieved depending on the number of runs of the algorithm with more than 10 runs improving and stabilizing the accuracy**
- Accuracy was measured by checking if the predicted value for a new FOMC minutes document was equal to the actual Fed rate decision – hike, no hike, rate cut
- **For Ngram matching the tokenized words were converted into ngram dictionaries**
 - The ngram dictionaries were created for each type of fed fund rate decision and contained the (ngram, its frequency) as the (key, value) pair
 - Following ngram types were used - Trigrams, fourgrams and fivegrams (continuous sequence of 3, 4, 5 tokenized/lemmatized words in the text)
 - Further random sampling of the ngrams was done for the ngram models as there are more no rate hike and rate hike ngrams compared to rate cut ngrams
 - A random sample of the excess ngrams equal to the number of rate cut ngrams was used
 - Predicting the fed fund rate decision for a new minute's document (not included in the ngram dictionaries) was done using the matching frequency of the ngrams
 - Frequency, normalized frequency, simple match count were used to predict outcome based on new document
 - Results were highly skewed to the no hike ngram dictionary as that is the largest given there are more no hike documents compared to hike and rate cut
 - ***This is not a good algorithm and requires more analysis to use better techniques***
- **Note** - Standard vectorization and related predictive models were not used as the new document to predict has a very small subset of the ngrams in the vector space. More analysis is being done to see if any other techniques/algorithms can be used (RNN, etc)

Results

Classification algorithms gave decent results (60%-80% match) while Ngram matching gave very poor/wrong results (highly skewed to no hike documents)

Useful Notes, Future Analysis

Note 1 - This is a very small sample of the entire set of documents published by the Federal reserve and we haven't covered the various speeches and comments they make at various official and unofficial events. This analysis covers a simple premise and is not sufficient for all possible variations to the relationship between the fed fund rate decision and the data in different documents/speeches which in turn depends on macroeconomic and other data over varying timeframes. There are many other variations that can be tried with the documents, speeches, data, time lag, different algorithms and their parameters. Users can test that on their own and use as they see fit.

Note 2 – The model accuracy is variable and requires lot of training as it is quite difficult to get the context of words/ngrams in documents for text analysis using NLP. While humanly we are able to interpret things more accurately (though not always), for machines the complexity of language comes into play.

Reason for not using standard Fed statements for ngrams

The Fed has always used a very subtle and nuanced approach to convey its message in its statements, as that gives it more leeway in determining the desired path for fed fund rates in relation to its dual mandate of supporting the US economy through maximum employment and stable prices (inflation and interest rates). While they have been known to use some standard boilerplate text like “some further gradual increases” recently, they don’t cover the entire universe of all historic statements and can give a skewed analysis of document/ngram/word importance considering we are now entering a more uncertain/volatile economic environment. Monetary tightening after the historic low rates for a long time post 2008 credit crisis has impacted the economy and global economic slowdown is forcing central bankers to turn dovish. This means that the recent standard statements might not give enough and accurate insight into future rate path, as there is a possibility of rate cuts. In light of this it makes sense to use a larger pool of documents randomly and analyse that using various well known NLP algorithms.

Future analysis possibilities – Other Countries Central banks statements

This approach can be used for statements made by other central banks like EU, UK, India etc as they too take similar approaches to interest rate decisions and how they convey it. They have their own wording nuances and history in relation to their economies and global relevance.

Future analysis - Combine with macroeconomic data to predict currency prices

The results from this analysis can be combined with the macroeconomic analysis done earlier to predict currency prices - ***Results of running multiple linear regression and random forest algorithms on macroeconomic data to predict currency prices (3 Apr '18)***. This is very useful as both central bank interest rate decisions and macroeconomic data together have the biggest impact on currency prices. Combined together, macroeconomic and central bank data can be used to build a more robust algorithm for predicting currency prices. This is being done for major currencies like USD, EUR, GBP and INR and the results will be published soon.

Sample code and Data

Data used for this analysis along with sample code is given in the below Github location <https://github.com/mobicloudtrees/CentralBankStatementsRateDecisionsNLPAnalysis.git>

- **Data files** – 134 Text files (1999-2018) with naming convention FOMCmmmyyStmt.txt.
- **Sample code** – FOMCPredictClassifierNBDT_9Feb19.py,
FOMCDecisionPredictNgramMatch_9Feb19.py

Make sure you point the file loader to the correct location of the data files on your local drive.

Please use that data and sample code keeping in mind the disclaimer below.

Please get in touch if you see any errors or want to discuss this further at nitin@currentiax.com

DISCLAIMER – VERY IMPORTANT

The sample data and code are provided only for reference purposes and their accuracy or validity cannot be guaranteed. No guarantees can be made about the accuracy of the data and all data and analysis should be used for reference purposes only.

Users should carry out their own data collection, validation and cleaning exercise.

Similarly, they should carry out their own analysis by using different algorithms and varying their parameters as they see fit.

Please see our website for more data and analysis – www.currentiax.com

Please see the disclosure page on our website before reading this analysis - http://www.currentiax.com/#/aboutus/disclosure_about.