IFC-Bank of Italy Workshop on "Data Science in Central Banking: Applications and tools"

14-17 February 2022

A machine learning approach to narrative retrieval in economic news: the case of oil price uncertainty[1] [2]

Donald Jay Bertulfo,
Delft University of Technology & Asian Development Bank

[1] This contribution was prepared for the workshop. The views expressed are those of the authors and do not necessarily reflect the views of the Bank of Italy, the BIS, the IFC or the other central banks and institutions represented at the event.

[2] The author prepared part of this research while working at the Asian Development Bank (ADB). Work on index construction and evaluation benefitted from close guidance by Dr. Abdul Abiad and Dr. Irfan Qureshi, extensive support from Dr. Elisabetta Gentile, all of which are ADB economists. Meanwhile, subsequent work on natural language processing was done under the close supervision of Dr. Rachelle Sambayan and Dr. Fredegusto David of the University of the Philippines-Diliman.

A Machine Learning Approach to Narrative Retrieval in Economic News

The Case of Oil Price Uncertainty

Donald Jay Bertulfo

Delft University of Technology

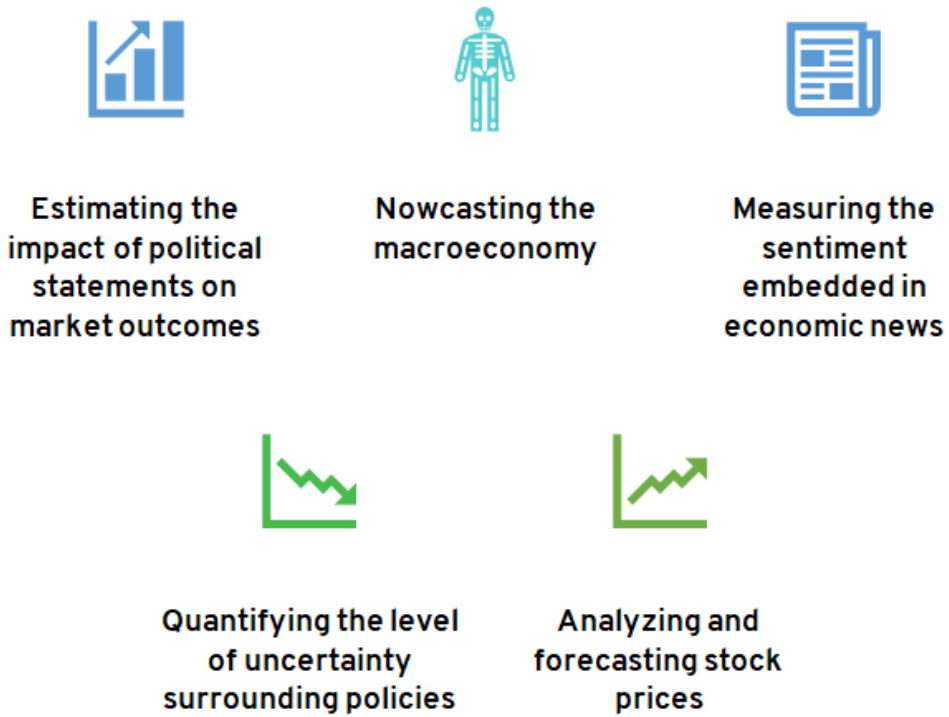
February 17, 2022

Overview

1. Introduction and Motivation
2. Literature Review and Contributions
3. Methodology
   - Text mining and document feature extraction
   - News-based OPU index construction
   - Text preprocessing and cleaning
   - Generating document embeddings
   - Measuring pairwise article similarities
   - Detecting article communities
   - Recasting document labels via NPMI
   - Topic Modelling
4. Results
5. Conclusion and Recommendations

Text as Data

Emerging research interest in economics: The analysis of large volumes of texts to uncover patterns in economic dynamics. Recent NLP applications include:



Analysis of economic shocks

The status quo: Pursued from a model-based, deterministic view of the macroeconomy

Our argument: The nature and magnitude of economic shocks are context-specific and temporally heterogeneous

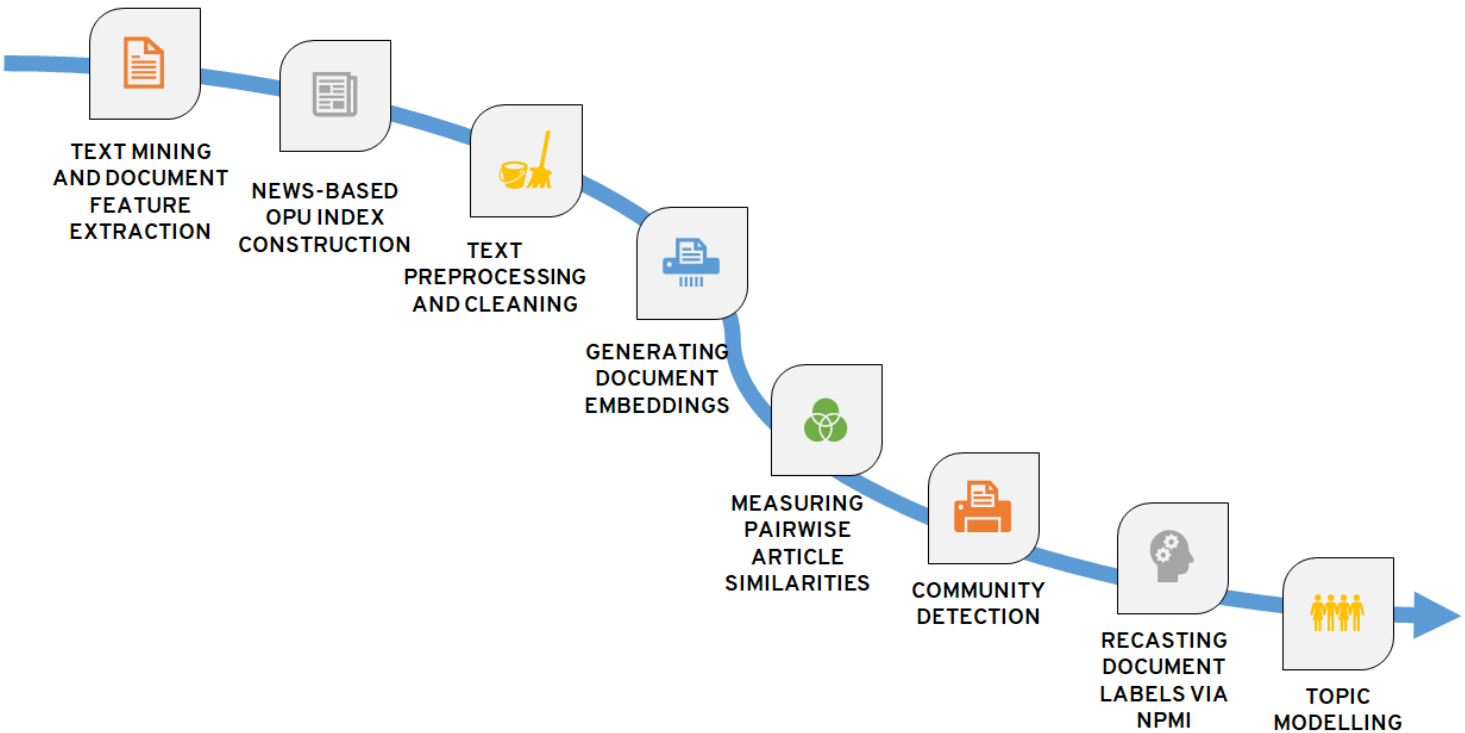
This research: aims to shed light on this complexity by pursuing the question from a data-driven, inductive and grounded approach

Extract salient themes that are common across episodes of economic shocks or crises

Retrieve salient narratives underlying shock episodes based on the content of talk in those moments

Analysis of economic shocks

Taking oil price uncertainty as a case study,



Literature Review and Contributions

- Linguistic information contained in texts reveals patterns that are useful in shedding light on the complexity of interactions among economic variables
- This research

contributes to literature that use news- or text-based methods to track economic movements

builds on applications of text-based dimension reduction techniques to extract salient themes in texts

extends literature on the use of neural networks to capture semantic regularities in text

Text Mining Procedure

News articles were mined from the Factiva search database, a Dow Jones news aggregator

The search algorithm takes articles

containing the trio of terms "oil or petrol or petroleum or gasoline," "gas", "price" and "uncertainty" (together with its synonyms)

which are no more than 2 words apart from each other

and with word count >99

+ restrictions on article type (not opinion, editorial, etc)

Downloaded full articles were stored in RTF format (100 articles per file)

Document feature extraction

- A rudimentary R code was used to extract important textual features such as title, date, publication, body of text and word count from each article.


THE ORANGE COUNTY
REGISTER

NEWS
Oil: Quake news sends market price up in Far East ← TITLE

Jonathan Lansner:The Register ← AUTHOR
155 words ← WORD COUNT
18 October 1989 ← DATE OF PUBLICATION
The Orange County Register ← NEWSPAPER NAME
OCR
English
Copyright 1989, Orange County Register. ← SEPARATOR

One immediate financial effect of the earthquake was that the market price for crude oil jumped nearly 29 cents a barrel -- or 1.5 percent -- in trading early this morning in Far East markets.

According to analysts, concerns that California pipelines and refineries may have been damaged by the San Francisco earthquake were behind the move. San Francisco Bay is home to at least five large refineries.

"It's not surprising," said Simon Flowers, an energy analyst at Ord Minnett Ltd., an Australian brokerage. "The West Coast is a major distribution point to other cities in the states as well."

Movements in crude oil prices often translate to **gasoline price changes** within a few days. For example, the Exxon Valdez oil spill in Alaska which shut down a major shipping lane for days pushed up pump prices in the spring.

Document ocr0000020011116dlai017x6 ← SEPARATOR

The New York Times

Financial Desk; 3
BUSINESS FORUM: FORECASTING INFLATION; 'P-Star' Can Spot Inflationary Trends

By L. DOUGLAS LEE; L. Douglas Lee is vice president and chief economist of County NatWest/Washington Analysis Corporation, part of the NatWest Banking Group.
881 words
22 October 1989
The New York Times
NYTF
Late Edition - Final
English
Copyright 1989 The New York Times Company. All Rights Reserved.

Figure: Sample Labeled Article Snapshot

The news-based OPU index

- Following Baker, Bloom and Davis (2016), the following standardization and normalization procedures were applied:
  1. For each month-year-newspaper, raw counts were scaled by the total number of articles.
  2. The month-year-newspaper level series was standardized to unit standard deviation from January 1969 to January 2020. Then, averages were taken across the 50 newspapers by month-year.
  3. Finally, the 50-newspaper series was normalized to a mean of 100 from January 1969 to January 2020.
- A sanity check was performed by plotting the OPU index across time points and identifying episodes in the history of oil price shocks which may help explain spikes in the OPU index

The news-based OPU index

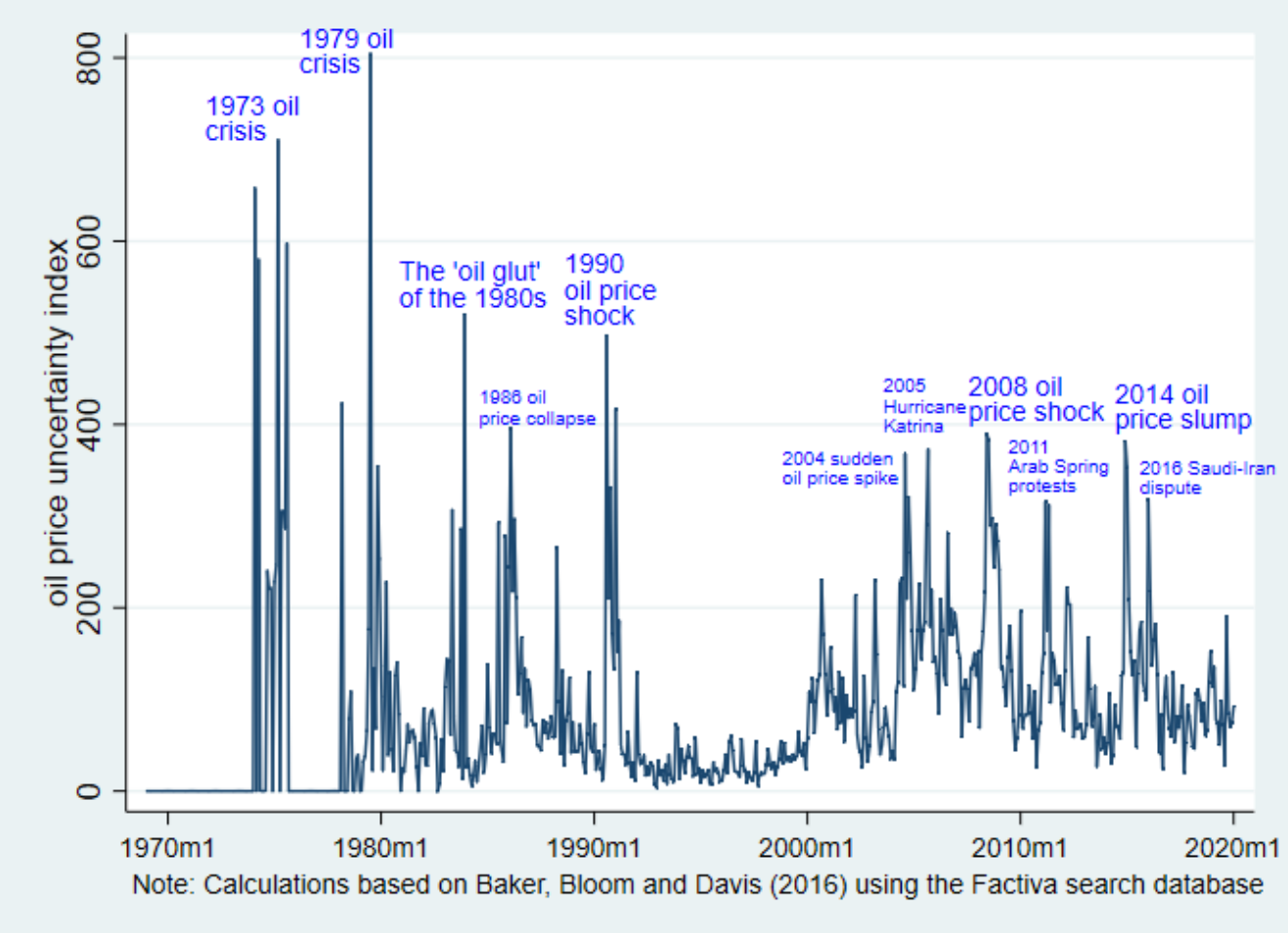


Figure: News-based global oil price uncertainty index, Jan 1969 - Jan 2020

Further validation

- The news-based OPU index was plotted vis-a-vis popular measures of oil price uncertainty to check for series co-movements. These measures include:
  - 60-day historical Brent volatility index (HVOLBREN)
  - 3-months implied volatility index (IVOLBREN)
  - 30-day volatility index (OVX)
- Correlation analysis suggests strong comovements between the news-based OPU index and other market-based measures of oil price volatility

Further validation

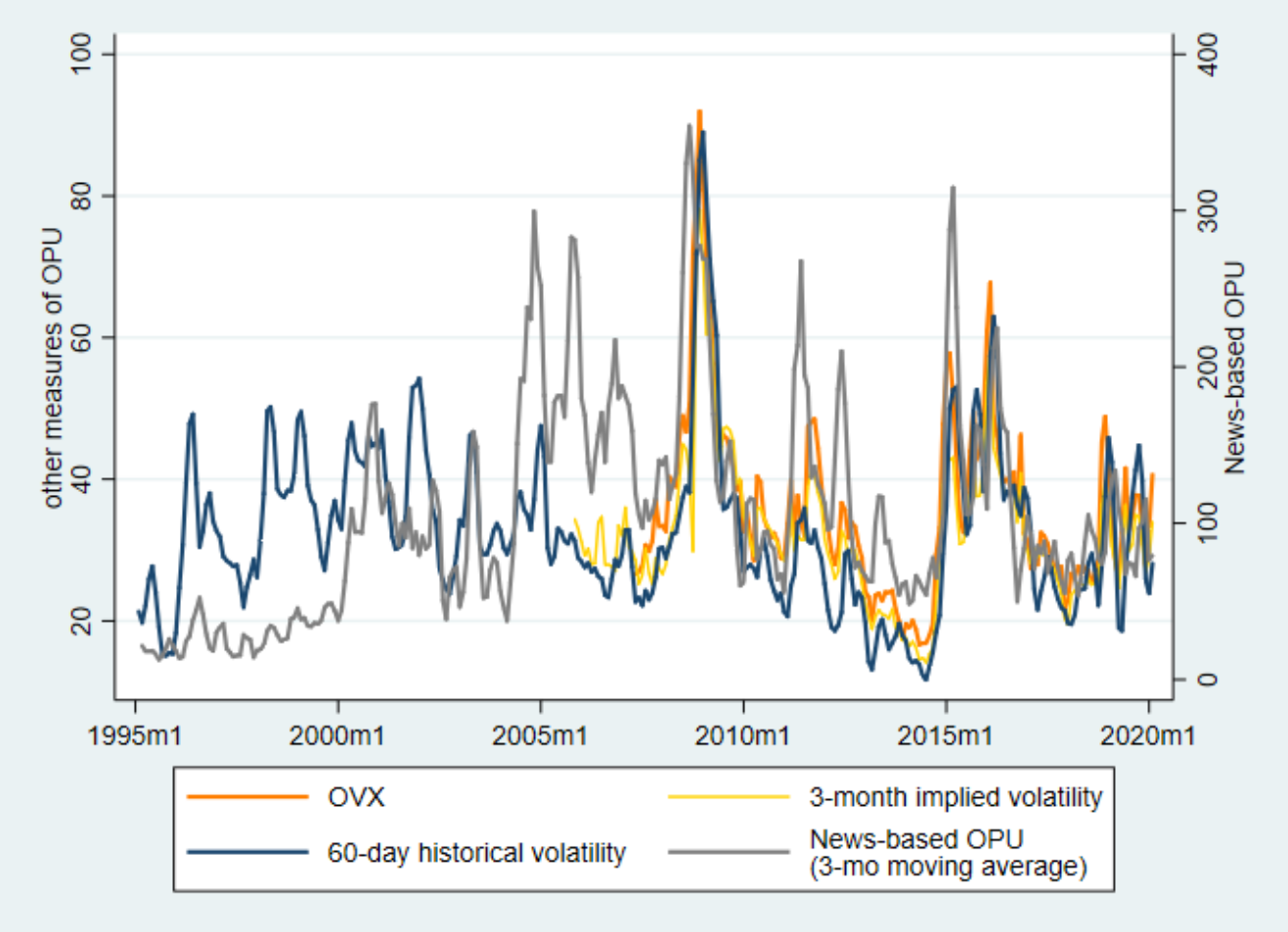


Figure: News-Based Global Oil Price Uncertainty (OPU) and Other Measures of OPU (January 1995 - January 2020)

Text preprocessing and cleaning

The following preprocessing procedures were implemented:

Tokenization

refers to the process of converting a document into a sequence of symbols called tokens. At this stage, each article was converted to unigrams (one-word tokens) using the tidytext package in R

Case folding and digit normalization

tokens were lowercased, punctuations and digits were removed from the corpus using regular expressions

Stopwords removal

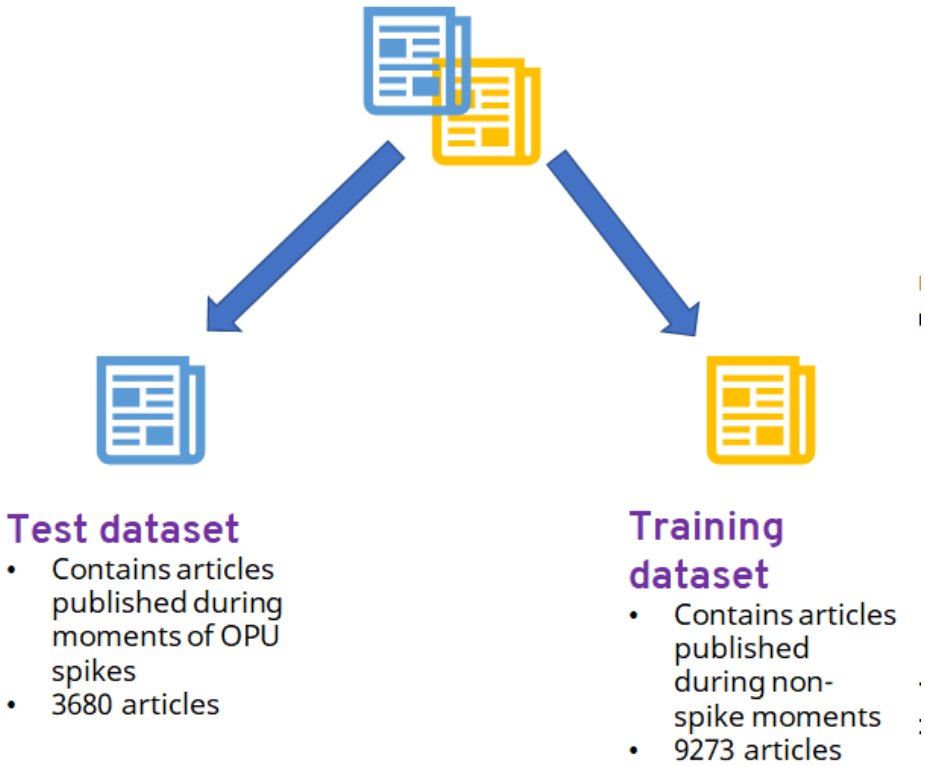
high-frequency words with low information content (e.g. 'the', 'is', etc) were removed. These stopwords are documented in the R 'stopwords' package

Text stemming

words were stemmed to their root using the Porter algorithm in R.

Text preprocessing and cleaning

- Further preprocessing involved reconstructing the cleaned and pruned articles (concatenation of word tokens).
- The entire text corpus was partitioned into two groups
- The cleaned and pruned training dataset was then fed into a doc2vec model (a type of paragraph embedding model)

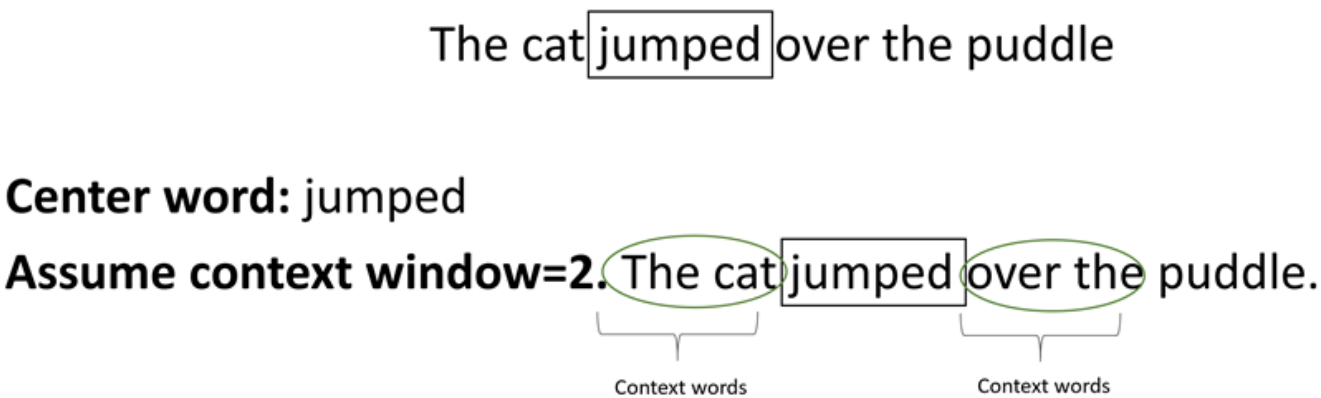


Document embeddings

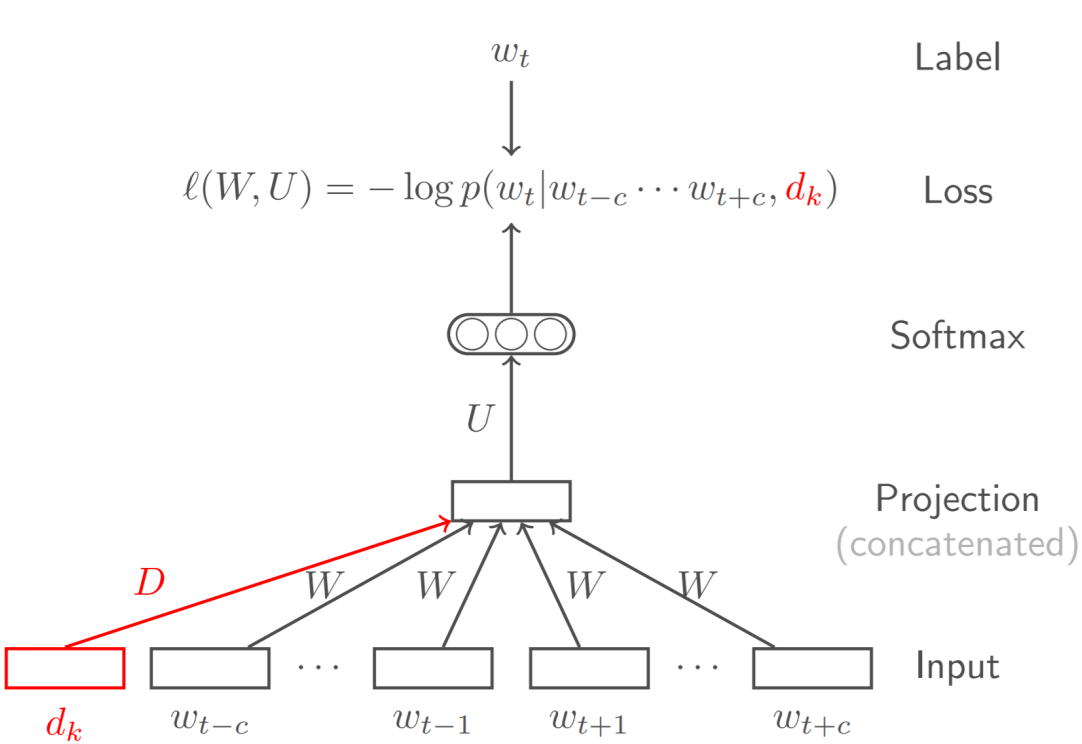
- **Distributional hypothesis**: words which occur in similar contexts tend to have similar meanings
- Ergo, documents that embed similar words tend to talk about the same things
- **Estimation Strategy**:
  - represent words as vectors;
  - input the word vectors, together with document labels into a neural network model;
  - model learns document semantics; pretrained model can be used to infer vectors for additional documents (i.e., articles in the test set)
- Here, a doc2vec model was implemented to learn vector space representation of documents

Document embeddings: the doc2vec models

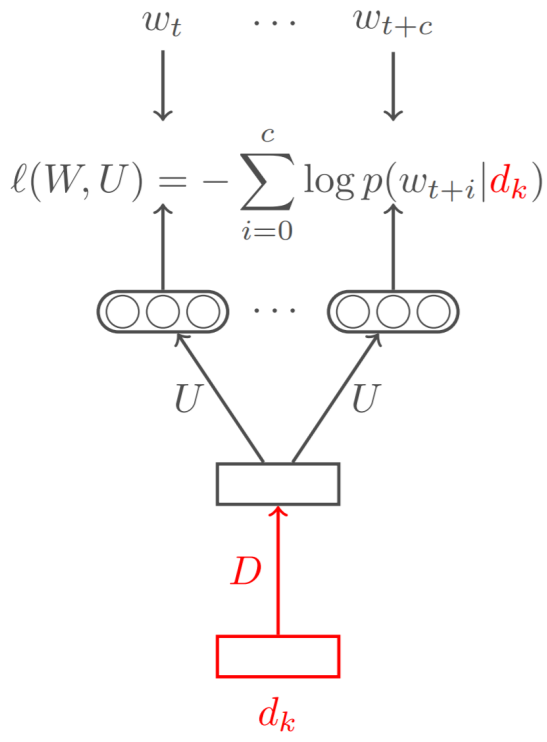
- Doc2vec models are unsupervised neural network models that generate fixed-length vector representations (also called "paragraph vectors") from variable lengths of texts
- Two types:
  - **PV-DM/Distributed memory:** learns to predict the occurrence of a center word, given context words and a document label.
  - **PV-DBOW/Distributed bag-of-words:** learns to predict context words given document label



Document embeddings: the doc2vec models



(a) PV-DM



(b) PV-DBOW

Figure: Doc2vec model architectures

Document embeddings: the doc2vec models

- The reconstructed cleaned and pruned articles in the training set were fed into the Doc2Vec DBOW model, using the following parameters: max epochs = 100, vector size = 300, alpha = 0.025, min alpha = 0.00025, number of negative samples = 5, window size = 5, sample = 0.001.
- After training, the pre-trained doc2vec model was used to *infer a paragraph vector for each document in the test corpus*
- This process yielded a matrix of size 3860 (number of articles in test set) x 300 (number of features in hidden layer). For our purposes, we call this matrix the **paragraph matrix**

Pairwise document similarities

- Given inferred paragraph vectors for each article in the test set, we are now interested in knowing **how similar the articles in the test set are**.
- To compute for semantic closeness irrespective of document length, we invoke the concept of cosine similarity

Cosine similarity

Given two nonzero vectors $\mathbf{x}$ and $\mathbf{y}$ of length n, cosine similarity (i.e., $cos\_sim(\mathbf{x}, \mathbf{y})$) is given by the formula

$$cos\_sim(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{xy}}{\|\mathbf{x}\|\|\mathbf{y}\|} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}$$

where $\|\cdot\|$ represents the Euclidean norm and $x_i, y_i, i = 1, 2, .., n$ are the components of $\mathbf{x}$ and $\mathbf{y}$ respectively.

Pairwise document similarity

- To estimate pairwise semantic closeness in the document space, cosine similarities were computed for each pair of paragraph vectors $\mathbf{d_i}$ and $\mathbf{d_j}$.
- Values were stored in a matrix $\mathbf{P} = [cos\_sim(\mathbf{d_i}, \mathbf{d_j})]$ to yield a symmetric matrix of pairwise cosine similarities (i.e., note that $cos\_sim(\mathbf{d_i}, \mathbf{d_j}) = cos\_sim(\mathbf{d_j}, \mathbf{d_i})$ for any i, j).
- Establishing a link between matrix and graph theory, $\mathbf{P}$ can also be regarded as the matrix representation of a weighted underlying similarity graph with documents as nodes

Community detection

- Taking **P** as an underlying matrix representation of a similarity graph with articles as nodes, community detection was performed by inputting this adjacency matrix into the Louvain algorithm.
- Louvain is a popular unsupervised clustering algorithm that detects graph communities based on the principle of modularity maximization.
- **Modularity** is a measure of the density of links inside communities as compared to links between them.

Community detection

Modularity

For weighted networks, modularity is given by:

$$M = \frac{1}{2W} \sum_{i,j} \left[w_{i,j} - \frac{k_i k_j}{2W} \right] \delta(c_i, c_j)$$

where $w_{i,j}$ refers to the weight between edges i and j, k_i is the sum of the weights of links that connect to node i, W is the sum of all the links in the network, c_i refers to the community where node i belongs and $\delta(c_i, c_j)$ is a function that takes the value of 1 if $c_i = c_j$ and 0 otherwise.

Community Detection: The Louvain algorithm (Phase 1)

1. Assign each node i to a distinct community C_i.
2. For each node i, do the following:
   1. Identify the set of neighbors J of i
   2. For each neighbor $j \in J$, evaluate the gain in modularity that would take place by removing i from its community to C_j. The change is given by the following formula:

$$\Delta M = \left[\frac{\sum_{in} + 2k_{i,in}}{2W} - \left(\frac{\sum_{tot} + k_i}{2W}\right)^2\right] + \left[\frac{\sum_{in}}{2W} - \left(\frac{\sum_{tot}}{2W}^2\right) - \left(\frac{k_i}{2W}\right)^2\right]$$

   where $\sum_{in}$ is the sum of the weights of the links inside community C_j, $\sum_{tot}$ is the sum of the weights of links to all nodes in C_j, $k_{i,in}$ is the sum of the weights of links from node i to node C_j
   3. Move node i to the community with the biggest modularity gain. If no positive gain is found, i stays in its original community
3. Repeat the process until no further improvement can be achieved

Community Detection: The Louvain algorithm (Phase 2)

1. Construct a new network whose nodes are the communities found in Phase 1. In this new network, the weights of the links between the new nodes are given by the sum of the weights of all links between the nodes in the corresponding communities. Weighted self-loops reflect links between nodes of the same community.

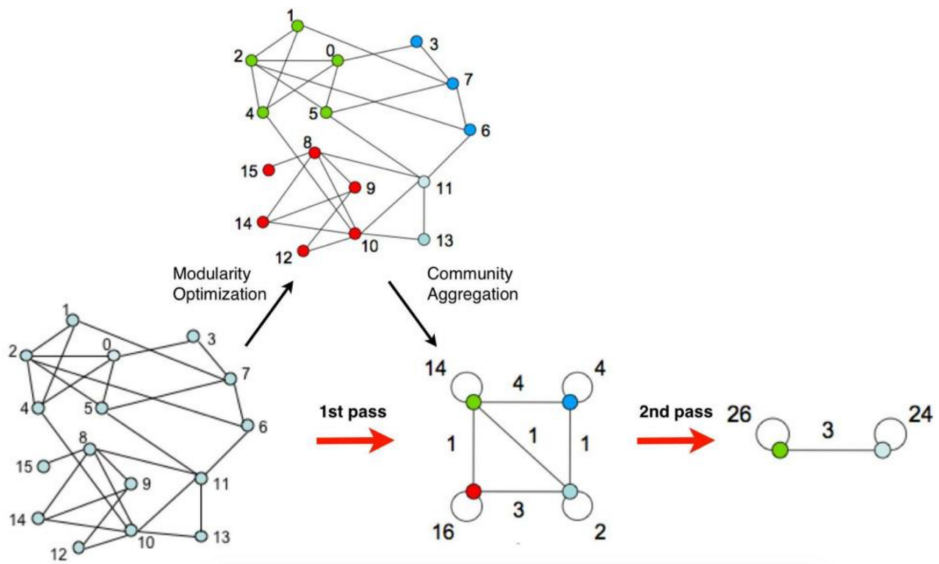


Figure: Louvain algorithm, a visualization

Community Detection: Implementation

1. **P** is then fed into the Louvain algorithm (network weights correspond to entries of **P**), which was ran in R via the igraph package.
2. The algorithm yielded a category label for each article $d_n \in \mathcal{D}_{test}$ wherein the category label reflects the community membership of d_n.
3. **Three document/article communities were detected by the Louvain algorithm.** To explore the semantic structure of these communities, the articles were grouped by community membership and visualized using wordclouds.

Sanity check: Wordclouds

Figure: Community 1 Wordcloud

Sanity check: Wordclouds

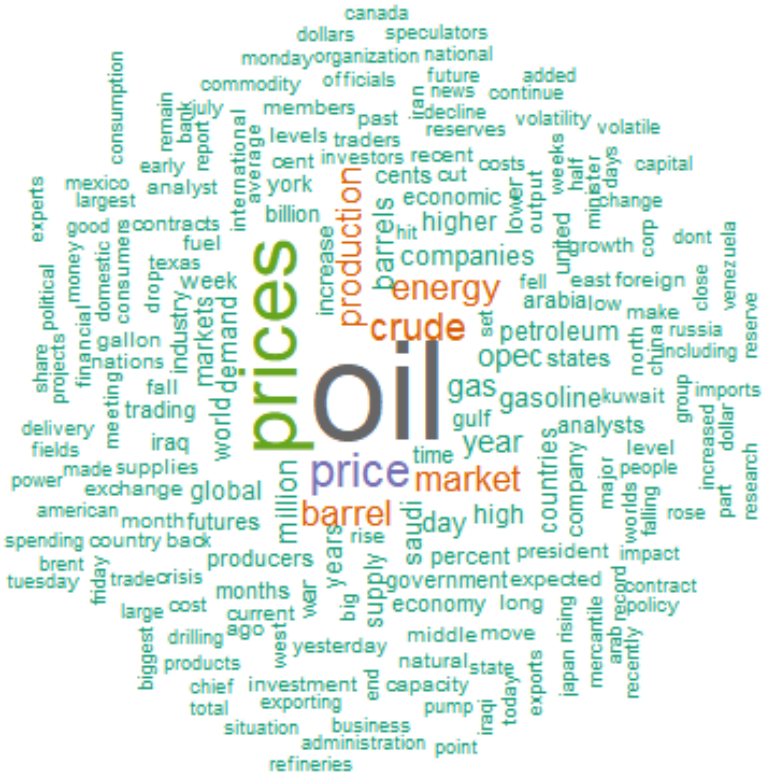


Figure: Community 2 Wordcloud

Sanity check: Wordclouds

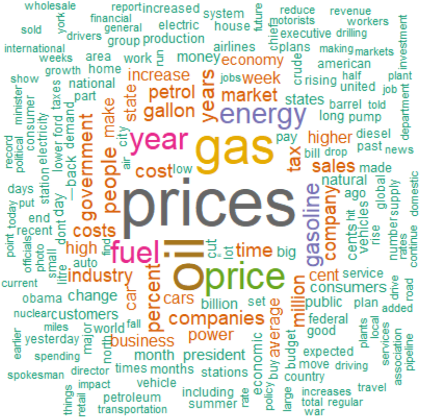


Figure: Community 3 Wordcloud

Recasting document labels via NPMI

1. Constructing a word-by-community co-occurrence matrix
2. Computing for normalized pointwise mutual information

$$NPMI(\mathcal{W}_i, C_k) = \frac{log(\frac{p(\mathcal{W}_i, C_k)}{(p(\mathcal{W}_i)(p(C_k)})}{-logp(\mathcal{W}_i, C_k)}$$

$p(\mathcal{W}_i)$ pertains to the relative salience of word $\mathcal{W}_i$ in the entire corpus, $p(\mathcal{C}_k)$ denotes the share of community C_k in the total number of articles in the entire corpus and $p(\mathcal{W}_i, C_k)$ represents the probability that word $\mathcal{W}_i$ and community C_k co-occur.

3. Computing for summary scores

$$S_{i,k} = NPMI(\mathcal{W}_i, C_k) - \sum_{n \neq k} NPMI(\mathcal{W}_i, C_n).$$

for each unique word $\mathcal{W}_i$ in the corpus

4. Aggregating summary scores
5. Revising community labels of each article in $\mathcal{D}_{test}$ according to the following decision criterion: $\bar{\mathcal{C}}_k(d_n) = \max_k \mathbb{E}_n[S_{i,k}]_{\mathcal{W}_i \in V}$

Topic Modelling

- A **topic** is a "distribution over a fixed vocabulary"
- Topics consist of words, which are spread out in texts
- The goal of topic modelling is to discover, in an unsupervised manner, latent themes from documents
- Topic extraction using latent Dirichlet allocation was implemented using the Mallet package in Python
- Using regression analysis, we identify which among the topics extracted were most associated with which document community

Topic Classification

Topic	Top keywords	C1	C2	C3
1	market, share, pc, price, group, year, time, ftse, bank, oil	✓		
2	compani, share, deal, offer, million, sell, valu, bid, plan, board	✓		
3	war, iraq, kuwait, crisi, natio, gulf, oil, presid, middle_east, russia		✓	
4	gas, energi, natur, year, price, oil, state, texas, pipelin, drill			✓
5	year, job, busi, work, worker, cut, incom, small, pay, budget			✓
6	energi, electr, power, heat, fuel, cost, home, effici, gas, system			✓
7	airlin, fuel, year, cost, carrier, fare, air, hedg, travel, flight			✓
8	time, peopl, good, thing, long, make, back, mani, lot, reason			✓

Topic Classification

Topic	Top keywords	C1	C2	C3
9	car, vehicl, sale, fuel, year, auto, truck, ford, gas, model			✓
10	cent, dollar, yesterday, currenc, euro, gold, close, european, week, lowe	✓		
11	oil, product, opec, price, produc, barrel, countri, saudi_arabia, world, market		✓	
12	uk, britain, pound, british, govern, cut, energi, warn, industri, brown			✓
13	canada, canadian, govern, today, billion, cent, year, report, busi, price	✓		
14	stock, index, market, fell, point, gain, rose, investor, close, share	✓		
15	presid, state, obama, republican, bill, feder, bush, congress, senat, admi			✓
16	price, gasolin, gas, gallon, station, averag, pump, refineri, week, day			✓

Topic Classification

Topic	Top keywords	C1	C2	C3
17	price, cost, increas, compani, custom, consum, rise, pay, higher, bill			✓
18	rate, economi, growth, inflat, econom, bank, year, interest_r, rise, econo	✓		
19	project, develop, industri, plant, build, power, plan, invest, product, busi			✓
20	price, year, sinc, drop, month, fall, week, expect, declin, lower			✓
21	time, show, page, univers, call, work, offic, art, polic, peopl			✓
22	market, oil, price, trade, futur, contract, trader, crude, specul, commod		✓	
23	market, invest, investor, fund, stock, bond, year, manag, money, equiti	✓		
24	percent, year, sale, month, report, retail, consum, spend, juli, increas	✓		

Topic Classification

Topic	Top keywords	C1	C2	C3
25	oil, price, barrel, crude, demand, energi, high, rise, higher, suppli		✓	
26	govern, elect, polit, world, meet, econom, leader, countri, issu, parti			✓
27	china, global, japan, world, economi, export, year, growth, econom, countri	✓		
28	govern, bank, india, market, sector, year, polici, increas, current, credit	✓		
29	oil, compani, industri, bp, explor, product, billion, shell, field, analyst		✓	
30	tax, price, petrol, fuel, budget, litr, govern, cost, increas, diesel			✓
31	citi, peopl, day, servic, area, road, transport, drive, home, car			✓
32	million, year, billion, compani, profit, quarter, share, earn, revenu, expenditur	✓		

Topic Salience During Shock Moments

- **1974-1975**: OPEC, economic growth, politics and jobs
- **1979**: oil demand/supply, innovation
- **1983-1985**: OPEC, natural gas
- **1986**: oil supply/demand, OPEC, decline in gas prices, speculation in oil prices
- **1990**: stock prices, oil demand/supply, war, speculation in oil prices
- **2004**: retail gasoline prices, consumer prices, stock market, economic growth, oil demand/supply, profits, consumption
- **2006**: retail gasoline prices, tax
- **2008**: travel, decline in oil prices, automotives, jobs
- **2012**: retail gasoline prices, American politics, natural gas, innovation
- **2015-2016**: decline in gas prices, share price

Conclusion

- Machine learning has immense potential to uncover underlying narratives behind economic shock episodes
- The talk about economic shocks is embedded in multiple, interlocking topics
- Context matters in policy!

Recommendations

- Hyperparameter tuning and sensitivity analysis
- Linking computationally-derived narratives with macroeconomic time series
- Apply the same narrative extraction technique to unpack stories underlying other types of shocks (e.g. COVID-19 pandemic, geopolitical conflicts)