Prediction of the crude oil price thanks to natural language processing applied to newspapers

Sophie Trastour, sophietr; Maxime Genin, mgenin; Arthur Morlot, amorlot

December 14, 2016

Abstract: This study presents how commodity prices can be related to newspapers. More precisely, we will focus in this paper on the crude oil prices. We used topic modeling (Latent Dirichlet Allocation) to extract major topics from news articles. As a result, our features were the daily and montly proportions of articles for each topic. Our output was the daily and monthly stock oil prices adjusted by the usd inflation. We tested different number of topics and preprocessed our dataset so as to end up with relevant topics. Then, we implemented different models to fit our data: linear regression, Theil Sen regression, ridged least square, locally weighted regression. We also introduced polynomial features. We got promising results, especially with the locally weighted regression using 60 topics trained over 90% of our dataset: we were able to follow the dynamics of the crude oil prices including peaks or sudden moves and our explained variance was equal to $R^2 = 0.70$.

Introduction

Nowadays, oil is one of the most important commodities in the world. It is used everywhere - from the plastic to the bitumen via many fertilizers and pesticides. It also has a strong influence on the way stock markets behave: it can trigger an economic crisis as well as push inflation in Europe or in the US. It has now became the center of many attention, and as a result, it is often discussed in newspapers.

In this project, we focused on the correlations that could exist between the stock and commodity prices of crude oil and external factors highlighted in newspapers. Indeed, most financial studies focus on numerical data in order to predict price movements because it is easier to manage and easily accessible. However, this process cannot take into account the macroeconomics events or even simple news that nevertheless impact a lot the stock oil prices.

Thanks to machine learning and NLP (natural language processing) techniques, more and more documents can be processed in a semi-automated way. In this study, we use topic modeling (Latent Dirichlet Allocation) to extract the main topics from the articles in newspapers such as New-York Times, Reuters and the Associated Press so as to predict the movement of the stock oil price.

Dataset

We used the NY Times API to get automatically all articles from 1986 to 2015 containing the words "oil price". It was not possible to get the whole text of the articles but most of the time, we were able to extract the headline, an abstract and a snippet. We also used several API, the `nytimesarticle` package, and the python function `time.sleep()` to avoid the limitations of 5 articles per second and 1000 articles per day. Eventually, we successfully downloaded around 32,000 articles. We also used the crude oil price data from the EIA website as well as the US inflation from the federal Bureau of Labor Statistics.

NLP methods

Several machine learning algorithms could be used to process newspapers articles. Two of them lead to good results: the `word2vec` approach, and the topic modeling one. A short literature review showed that topic modeling was the most appropriate, as it would allow us to learn the topics present in our documents, which is exactly what we need to know in order to predict the stock oil price.

Topic Modeling: Latent Dirichlet Allocation (LDA)

In this section, we will offer a quick review of an algorithm that allows topic modeling, before describing how we applied it to our problem. Following the ideas of [1], we decided to use a Latent Dirichlet Allocation technique, for its capacity to capture multiple topics within a document.

As the focus of this document is more on our methodology and first results, we will only give a brief description of the LDA algorithm. A complete one can be found in [1].

The intuition behind the LDA is the following: each document is a "mixture" of different topics. A document may be 90 % about "oil" and 10% about "cars" (see figure 1). We now make an assumption: if a document is composed of 90 % about "oil" and 10% about "cars", then it is constructed by randomly sampling 90 % of its words from a distribution about "oil", and 10% from a distribution about "cars" (the ordering of the words does not matter to the algorithm). We end up with three hidden variables to explain our corpus:

- The topics, that is to say the words distribution inside a topic
- Per-document topic distributions
- Per-document per-word topic assignments

Each of them can be set to a specific prior before we run the algorithm, to encode information known by humans about the subject.

The LDA then uses inference algorithms to compute the posterior on these distributions and infer the more likely ones. The inference techniques will not be described here but they can be found in [1].

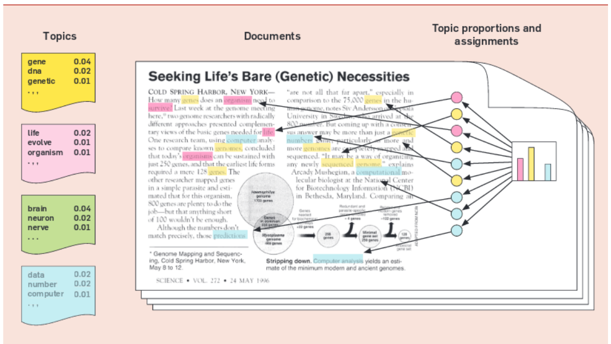
Topics
Documents
Topic proportions and assignments
gene 0.04
dna 0.02
genetic 0.01
...
life 0.02
evolve 0.01
organism 0.01
...
brain 0.04
neuron 0.02
nerve 0.01
...
data 0.02
number 0.02
computer 0.01
...
Seeking Life's Bare (Genetic) Necessities
Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.
SCIENCE • VOL. 272 • 24 MAY 1996

Figure 1: An explanation of topic modeling [1]

Preprocessing steps for LDA

There are packages available to do topic modeling in python. We used gensim, a package that trains LDA, and, pyLDAvis, a python library for interactive topic modeling visualization [3]. This latter shows the inter-topic distance calculated and projected using multi-dimensional scaling (MDS) with a Jensen-Shannon divergence distance. As we can see in figure 2, the weight of each topic is proportional to its size in the visualization: the bigger the circle depicting the topic, the greater the weight of this one in the dataset.

As previously described, we used articles from NY Times, Associate Press and Reuters. There were treated as a document and preprocessed by:

- setting to lower case
- removing the punctuation
- removing stop words (like “I”, “my”, “their”) which does not carry meaning
- lemmatizing the words (so that “like” and “liked” would be treated the same way)
- removing the numbers

The list of stop words and the data for lemmatizing words were extracted from the nltk package in python. We also added our own stop words that were common words irrelevant for our query on “oil price”. Once done, we learned the main topics of each article using gensim package.

When we first ran the LDA algorithm, we obtained irrelevant topics out of 10: one was containing common words and two were related respectively to “restaurant” and “art”. In order to get rid of the former, for each of the 10 topics, we printed the 50 more frequent words and remove the irrelevant ones by updating our stop words list. We then rebuilt the model. We interated this procedure twice so as to end up with reasonable topics (arount 50 words were added to the stop words). As far as the topics “restaurant” and “art” were concerned, resulting from the polysemy of “oil”, we removed the articles which main topic was one of them. We then retrained a model with the articles kept (around 3,000 articles out of our 32 000 dataset were removed with this procedure). The new model only contained topics that were of interest to our problem.

In the milestone report, we proposed to introduce a prior on the LDA, in order to improve the quality and relevance of the topics. After looking at the topics found, we did not see any topic that needed improvement: all the topics were sensible, and all the geopolitical and financial events that could impact the price of crude oil seemed to be present. Therefore, we did not proceed forward with the prior.

Resulting topics

Figure 2 shows the resulting topics after applying Latent Dirichlet Allocation algorithm and preprocessing the model as previously explained. The number of topics is set to 10 (this number will be discussed further). The topics are defined by their more frequent words, therefore, it is possible to find an interpretation for each of them:

1. Oil as a commodity
2. Oil production
3. Market finance
4. Macro economy
5. American Economy
6. Corporate finance
7. US Election and US Politics
8. Relations between China and the USA
9. Conflicts with Russia
10. Middle East conflicts

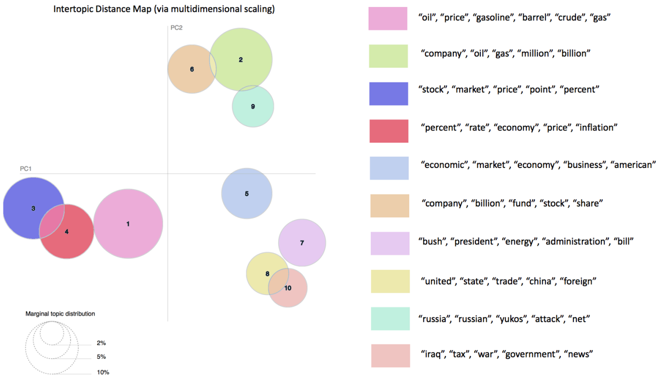

Figure 2: Multi-dimensional scaling visualization with respect to the two principle components of the topics found - Main words for each topic

Before any quantitative analysis, we can give a qualitative interpretation of the topics found. For example, there should be a correlation between the number of articles related to middle east conflicts and crude oil price: they usually mean less supply in crude oil and increasing prices. Similarly, conflicts with Russia should be positively correlated with the crude oil price whereas US election should be negatively correlated. The US election as well as the US political stance with respect to the oil producing countries may introduce uncertainty in the US demand of crude oil.

Features engineering and regressions

In order to make day by day predictions, we extracted the topics distributions of each article, and selected the main topic of this latter. Each article was therefore defined by a single topic. We then merged all the articles from the same day together. We ended up with the following dataset: for each day, we had the number of articles published, and their topic distribution. We did the same procedure on a monthly basis between 1986 and 2015.

In addition to that, we used two predicted variables: the stock oil price, p_i, which is the price at a day/month i, and the return at a day/month i, defined as $1 - \frac{p_i}{p_{i-1}}$. All returns and prices were adjusted to incorporate the dollar inflation from 1986.

Linear regression on daily features

- We first did a linear regression between the daily distribution of topics previously computed and the daily return. We obtained the following coefficients $10^{-4} \times [0.87, 4.8, -5.5, -9.1, 3.8, 2.9, 2.3, 12.2, -26.7, -5.5]$ and an intercept term of 0.0002. Figure 3 shows the results of our regression versus the real returns of the crude oil price for 500 days. As we can see, we do not predict much because the return is too volatile.
- Similarly, we performed a linear regression between the daily distribution of topics and the daily stock oil prices. We obtained the following coefficients $[5.2, -2, -1.7, -3.6, 0.03, 6.8, -1.3, 2.0, 2.4, 12.4]$ and an intercept term of 34.6. Figure 4 shows the regression versus the real prices for 200 days.

As we can see, daily returns and daily prices are too volatile to give good predictions. In addition to that, news articles usually have a longer term view. Varying the number of topics from 10 to 40, 60 or 100 did not give better results.

Linear regression on monthly features

- As the daily variation in prices can be very erratic, we performed the same analysis with monthly data. This led to a more robust model as the topic distributions is estimated from a greater number of articles. It also takes into account the fact that the stock oil prices do not necessarily vary on a daily basis. In figure 5, we plot the regression versus real returns. As we can see, the monthly return of the oil is still very volatile, and our regression does not capture it properly.
- However, linear regression of the monthly prices gives reasonable results (see figure 6). The coefficient of determination $R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$ is around 0.55 for 60 topics.

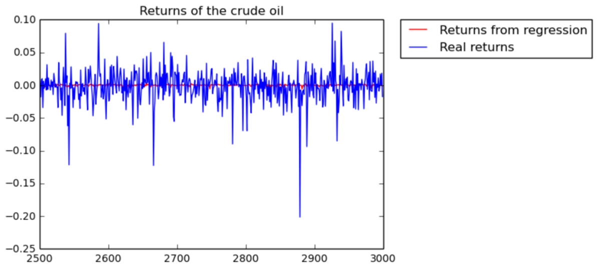

Figure 3: Daily version of returns from regression versus real returns

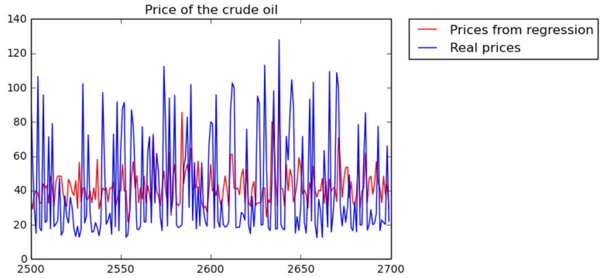

Figure 4: Daily version of prices from regression versus real prices

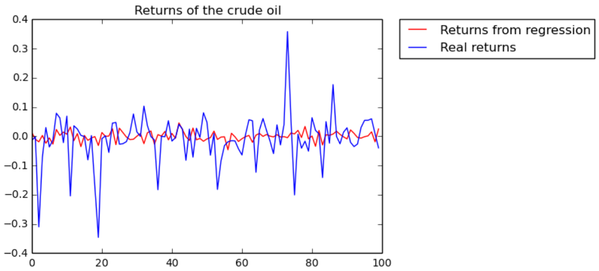

Figure 5: Monthly version of returns from regression versus real returns

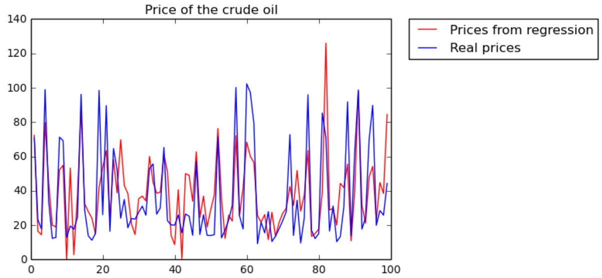

Figure 6: Monthly version of prices from regression versus real prices

Polynomial features

As the linear regression did not perfectly capture the prices variations ($R^2 = 0.42$), we used non-linear features to capture part of the volatility. We re-evaluated our models using all polynomials combinations of degree ≤ 2 of features. This was very efficient for all the LDA models that used 10 topics, and had therefore less than 100 polynomial topics. However, the more topics we added, the more features we had which leads to overfitting: the error on the training set was low but the error on the testing set was. We were no longer able to generalize our results. In this case, a feature selection should be made.

Other types of regressions

Because the crude oil price is quite volatile, attention must be paid to the weights of the features so as to avoid overfitting. In order to tackle this issue, we tested several models to find the best fit on our data: we used a Theil-Sen regression, a least-square regression and a ridged least square regression with a 10-fold cross validation strategy. Each model was tested over various number of topics (as training a LDA model is an expensive operation, we only tested these models with 10, 40 and 60 topics).

Theil-Sen regression presents two main advantages: it can be computed efficiently and it is insensitive to outliers. Our dataset had indeed outliers: some months were defined by an anormal number of articles as compared to other months. As a result, the input vectors for those months were far away from most of the training points in a L_2 distance. This could have a huge impact on the coefficients of the linear regression, leading to large errors. This is the reason why we fitted our data using a Theil-Sen regression. Contrary to what we expected, this latter did not performed as well as the linear regression ($R^2 = 0.42$ for 10 topics, and $R^2 = 0.55$ for 60 topics).

In order to avoid overfitting, we also used a ridged least-square regression which adds a penalty term to the cost function. This method slightly improved our results, especially when the number of features was important (100 topics or polynomial features).

Finally, we used locally weighted regression in order to introduce some non linearity. The idea was: according to some events, the crude oil price might follow a different model than as compared when there is no major event. Therefore, by using locally weighted regression, we would have a different model, for example, when a lot of articles are about the Iraq war (and therefore the price of oil is mainly controlled by this topic) and when most of the topics are on corporate finance (where each topic is important). This was the best model we found and we achieveed a coefficient of determination R^2 equal to 0.70. In the two figures below, we can see that the model follows very closely the real prices of crude oil. On average, the model underpredicts the price (which would not be an issue in case this process is implemented as a trading strategy: the most serious issue would be to wrongly predict a price too high).

The table below summarizes the coefficients of determination we obtained for the following models: linear regression, linear regression with polynomial features, Theil-Sen regression, ridged least square and locally weighted regression. We only reported three LDA models: 10 topics, 40 and 60 topics.

Models:	Linear	Linear + polynomial features	Theil-Sen	Ridged LS	Locally weighted
10 topics	$R^2 = 0.46$	$R^2 = 0.42$	$R^2 = 0.42$	$R^2 = 0.41$	$R^2 = 0.55$
40 topics	$R^2 = 0.50$	$R^2 = 0.51$	$R^2 = 0.48$	$R^2 = 0.49$	$R^2 = 0.62$
60 topics	$R^2 = 0.58$	$R^2 = -16$ (overfit)	$R^2 = 0.55$	$R^2 = 0.54$	$R^2 = 0.70$

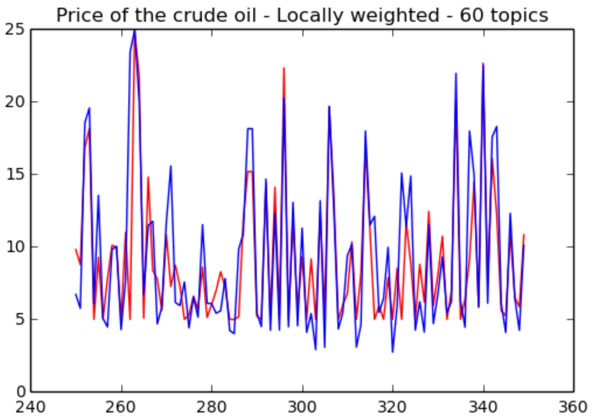

Figure 7: Monthly predicted and real prices vs time - Weighted regression

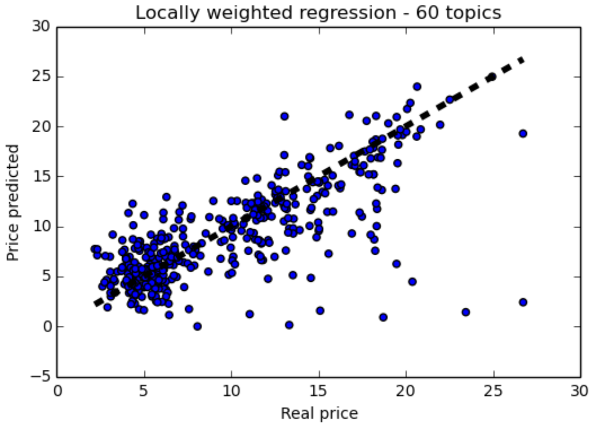

Figure 8: Predicted prices vs real prices - Weighted regression - 60 topics

Conclusion and future works

Topic modeling using Latent Dirichlet Allocation in addition to some preprocessing steps enables us to explore a large volume of news articles and understand the factors that could impact the stock and commodity prices. We successfully computed the main topic for each article so as to create monthly features - representing the distribution of the topics among the articles for this month. These latters were then used to match the inflation-adjusted monthly crude oil price.

The model that fitted the best this dataset was a locally weighted regression performed with 60 topics: it gave a coefficient of determination of 0.70. More qualitatively, we were able to capture the dynamic of the stock oil price and its volatility.

The coefficients of the regressions we performed also gave us useful information that corroborates the main opinion about relationships between some particular topics and the crude oil price: for example, a large number of articles about middle east conflicts would correspond to an increase in oil prices in the next month. The same analysis can be done for topic such as the relationship between USA and China, the US election, the economy in the US.

Although the results are already promising, taking into account that any correlation between articles and the crude oil prices has not been found earlier, there are several ways to improve this model:

- Add new features: topics distribution for oil and gas companies (Shell, Exxon, Chevron, ...), trend of the market
- Perform feature selections on the different topics to only keep the most relevant ones.
- Implement the model in an online fashion: determine the update frequency of the model according to new press releases
- Combine the model with a trading strategy using reinforcement learning

References

[1] Blei, David M. Probabilistic topic models. Communications of the ACM, 2012, vol. 55, no 4, p. 77-84.

[2] Newman, D. J., & Block, S. (2006). Probabilistic topic decomposition of an eighteenth-century American newspaper. Journal of the American Society for Information Science and Technology, 57(6), 753-767.

[3] Sievert, C., and Shirley, K.E., 2014, LDAvis: A Method for Visualizing and Interpreting Topics: Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces, p. 63-70