Forecasting Bulk Agricultural Commodities based on Baidu Index

Wenlong Ge , Yucong Zhang
{gewl, zhangyc}@shanghaitech.edu.cn

Abstract

In this paper, we will show that using Baidu Index are strongly related to the volatility of the bulk agricultural commodity price in China. And by introducing certain related Baidu Index feature, the accuracy of the prediction for the bulk agricultural commodity price can be improved greatly. In order find related keywords, we catch some data from a forum and build our own corpus, then we train a Word2vec model to calculate the cosine-distance. To tell which keyword may have strong causal relationship with the certain commodity price, we use both the ADF test and the Granger Causality Test to filter the keywords, and we implement a Python crawler to catch the Baidu Index for those keywords. In the end, we use a SVR model for experiment, and we prove that the prediction model with Baidu Index performs better than the model without it.

1 Introduction

In recent years, international commodity prices due to the global economic situation uncertain, the uncertainty of monetary policy, the impact of market speculation intensified, and other factors, showing more intense volatility. Taking into account commodity price itself has been the influence of multiple factors such as supply and demand, fundamental factors, exchange rate industrial output, interest rate speculation activities, and emergencies are difficult to achieve accurate prediction of its price. At the same time, economic globalization, commodity marketization and the continuous development of electronic information technology make the market thereby increasing interaction between further increased the difficulty of forecasting. At present, the domestic research on the commodity market is mainly based on traditional statistical data and related economic indicators, such as cointegration, GARCH model, ARIMA model, and VAR, to analysis and prediction. Therefore, we want to improve the accuracy of our futures market price predictions by introducing statistical data from the Internet as an indicator of external factors.

Internet search behavior can be interpreted as a measure of revealed expectations. Presumably, people search for information on topics they want to learn more about, or about things that are causing them concern. For example, if someone is not feeling well, they might search for information about the health symptoms they are experiencing. Similarly, if someone is concerned about managing their household expenses and believes the general price level is rising or that prices may soon rise, they might search for information about inflation. If someone is not worried about the prospect of rising prices, then they probably would not search for information about inflation, since most people are more fearful of inflation than of deflation. In an economy where wages are more sticky than prices, it is understandable that consumers would harbor anxiety about inflation. Therefore, we can surmise that if search query volumes on search engines for the term "inflation" are increasing, then the public is feeling increasingly anxious about the prospect of rising prices. Hence, changes in the volume of search queries about inflation can be interpreted as a measure of the general public's revealed expectations for future changes in the price level. Internet search activity data has been widely used as an instrument to approximate trader attention in different markets. This method has proven effective in predicting market indices in the short-term.

1.1 Related Works

Early applications of analyzing metadata from internet search queries and social media have been in the field of epidemiology and symptom surveillance of diseases. [Ritterman *et al.*, 2009] use data mined from Twitter posts to demonstrate that the public's expectations regarding a future event (i.e., a swine flu pandemic) can be extracted from social media data and those public perceptions can be used to reduce the forecast error of implied probabilities obtained from prediction markets where participants wager on the likelihood of a future event. [Guzman, 2011] shows that Google Trends can be used to predict inflation rate. [Basistha *et al.*, 2015] proved that it is possible to forecast commodity price volatility with internet search activity. [Afkhami *et al.*, 2017] indicates investor attention is widely reflected in Internet search activities and demonstrate search data for what keywords best reveal the direction of concern and attention in energy markets.

1.2 Our Contribution

No previous related work has explained the establishment of their keyword database. Based on this, we propose to use

natural language processing to train and extract keywords that are highly relevant to the type of futures we want to predict, which constitutes our keyword database; moreover, for them to operate further build our Internet search index. On the one hand, we use the Internet search index to predict the futures price process is more operable; on the other hand, we make the filtered keywords more interpretable.

Hence, our **goals**:

- Find the relationship between all kinds of agriculture commodity price and internet search activity and test them.
- Build a model to forecast various agriculture commodity price with internet search activity.
- Find the interaction among agriculture commodities and make the model more accurate.
- Test the result of our model and compare with other known model to prove our model is better.

In this paper, we will build our own Internet concern indicator based on the Baidu Index, and test it with an SVR model to compare whether this indicator will help the prediction of futures prices. The remainder of this paper is as follows: Section 2 describes the data used. Section 3 explains the methodology. Empirical analyses are presented in Section 4. And Section 5 concludes with a summary of the findings and future work.

2 Data

In order the analyze the predictive power of Internet search activity data on volatility of prices, we begin by gathering data. This section provides a description of the Baidu Index and the data processing.

2.1 Data Resources

Commodity Price

In this project, we have to use the bulk commodity price and Baidu index data. We get the commodity price data from Shanghai Stock Exchange(SSE). The data contains both daily exchange data and minute exchange data, which are exchanged in Dalian Commodity Exchange(DCE) from 2017 to 2019. In addition, we get the daily exchange data in DCE from 2011 to 2018 through the website[1].

Internet Concerns

We use Baidu Index[2] as the reflection of the internet concerns. Since we want to get the internet search activity in China, we decide to get this data from Baidu. Baidu is a very popular search engine in China, and due to its popularity, we believe that the internet search activity that is shown on Baidu can, to some extent, tell the amount of people who care about some particular commodity in China. Moreover, since people in China can hardly use google, they use Baidu most time as their first searching engine.

[1]DCE: http://www.dce.com.cn/dalianshangpin/xqsj/lssj/index.html

[2]http://index.baidu.com/v2/index.html

2.2 Data Processing

We performed the same calculations on commodity prices and Baidu Index. Daily rates of returns for each day is calculated by taking log differences in the daily spot prices:

$$r_d = \ln \frac{p_d}{p_{d-1}}$$

where r_d and p_d respectively represent return and price at day d.

3 Methods

3.1 Related Keyword Extraction

Given a word, we have to find the related keyword, which would better be under the financial context. For example, if we want to get the related keywords of 'Palm Oil', we do not want the keywords to be some sentence such as 'What's good about Palm Oil'; we would like the keywords to be professional, for instance 'Corn', 'Soybean Oil' and so on. In order to do this, we uses *Gensim*, which is first put forward in [Řehůřek and Sojka, 2010] along with our own corpus.

Gensim is a free Python library designed to automatically extract semantic topics from documents, as efficiently and painlessly as possible. The algorithms in Gensim, such as Word2Vec, FastTex, etc.

Build the Corpus

We catch down the data from a forum[3] which has tons of discussion about the future market. Since the website do not take any defensive measure, we simply catch the data by a Python package *urllib*. We have caught 14,905 messages on that forum related to the future market. To further enlarge our corpus, we add 10,000 financial news samples from *SmoothNLP* into our corpus.

Word2vec Modelling

Word2vec, which is first presented in [Le and Mikolov, 2014], is a group of related models that are used to produce word embeddings. These models are shallow, two-layer neural networks that are trained to reconstruct linguistic contexts of words. Word2vec takes as its input a large corpus of text and produces a vector space, typically of several hundred dimensions, with each unique word in the corpus being assigned a corresponding vector in the space. Word vectors are positioned in the vector space such that words that share common contexts in the corpus are located in close proximity to one another in the space. Especially, this proximity can be quantified by computing the cosine-distance of given vectors and the value is simple to compute. Hence, based on the corpus we have in 3.1, we can put it into Word2Vec, to let vectors represent the words.

When the vectors are created, we choose the top 20 keywords that has the biggest similarity with the selected keyword.

[3]http://bbs.jrj.com.cn/

3.2 Baidu Index Crawler

Once we have the keyword, we construct a Python crawler to catch the Baidu index for that keyword, as it is shown in fig.1.

Decryption The decrypt function has long been published by other hackers on the internet[4], and since the function can still be used, we decide to use it in our crawler.

sleep function In order not to be detected and stopped by Baidu, we implement a sleep function. Specifically, each time after the crawler request for data, it holds for a while(in our setting, we let the interval lying between 50ms to 90ms randomly).

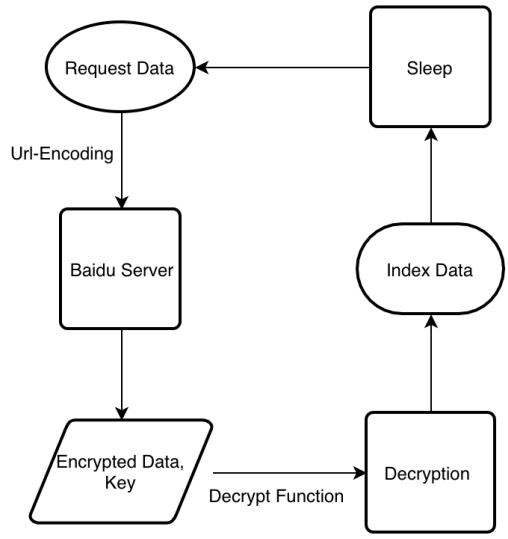

Figure 1: Index Crawler

3.3 Causality Test

ADF Test

To test whether one time series is stable enough, we use ADF unit root test to examine it. If there is a unit root, in other words, the null hypothesis is true, the time series is not stable. There are several criterion resulted from this, pvalue and critical values.

First, pvalue should not be too large, particularly, they must be as close as possible to zero. Typically speaking, when a value is smaller than 10^{-3}, it can be considered as a value close to zero. If the pvalue is above a critical size, we cannot reject that there is a unit root.

Second, the critical values. Critical values for the test statistic at the 1%, 5%, and 10% levels. If the output of the ADF test is smaller than all the values among those levels, we can reject that there is a unit root. If the output of the ADF test is bigger than only the value at the 1% level, but smaller than other two, we can also reject that there is a unit root. (Although there are chances that the null hypothesis is true, the probability of this is pretty small, we consider it as a false.) Otherwise, there is a unit root, thus the time series is not stable and cannot feed to a regression model.

[4]https://blog.csdn.net/weixin_41074255/article/details/90579939

Granger Causality Test

Aiming to test whether one time series is useful in forecasting another, we use *Granger causality test* to examine it. A time series X is said to Granger-cause Y if it can be shown, usually through a series of t-tests and F-tests on lagged values of X (and with lagged values of Y also included), that those X values provide statistically significant information about future values of Y.

Granger defined the causality relationship based on two principles:

(1) The cause happens prior to its effect.

(2) The cause has unique information about the future values of its effect.

In this paper, the time difference correlation analysis to determine each keyword leader (or consistent, lag) period and the correlation coefficient with the target variable. Time difference correlation analysis is a common correlation analysis to confirm the leading, consistent or lagging relationship between two economic variables. Its expression formula is as follows:

$$r_l = \frac{\sum_{t=1}^{n}(x_{t+l}-\bar{x})(y_t-\bar{y})}{\sqrt{\sum_{t=1}^{n}(x_{t+l}-\bar{x})^2\sum_{t=1}^{n}(y_t-\bar{y})^2}},$$
$$l = 0, \pm 1, \pm 2, \cdots, \pm N$$

where r_l denotes the correlation of period l; y denotes target variable, $\bar{y}$ denotes the average value of y; x denotes the keyword search volume, $\bar{x}$ denotes the average value of x; l denotes the leading (lag or consistent) period of the keyword.

We conducted Granger causality test on all selected keywords through ADF test. For the daily rates of Baidu index and returns for each day between January 04, 2011 and December 28, 2018, the following Vector Autoregression models are constructed:

$$y_t = c + \sum_{i-1}^{p}\alpha_i y_{t-i} + \sum_{j-1}^{q}\beta_j x_{t-j} + \epsilon_t.$$

where c is the constant coefficients;, and t is the day. p, q are the lag orders, and α, β are the coefficients of X and Y, and ϵ_t is the error term. Lag length all selected the best performance. The null hypothesis (H_0)that X_t does not Granger Cause Y_t is tested using the F-test. In other terms:

$$H_0 : \beta_j = 0,\ j = 1, 2, \cdots, q$$

Based on this setup, the rejection of null-hypothesis for keyword y can be considered to Granger cause X_t.

3.4 SVR Model

Support vector machine

Support vector machine (SVM) is a linear classifier first proposed by Cortes and Vapnik in 1995 [Buitinck *et al.*, 2013]. After introducing the kernel technique, it becomes a nonlinear classifier, which can well solve the problems of small samples, non-linear and high-dimensional models. Identify and apply it to other machine learning problems such as function fitting. When it is used for time series function regression, it is called support vector regression (SVR), which has

excellent and stable prediction ability for nonlinear time series. SVR maps samples to high dimensions through the change of non-linear functions. Feature space In this high-dimensional feature space, a linear function f can be found that can accurately express the correlation between output and input data, that is, the SVR function is:

$$f(x) = \boldsymbol{w}^T \phi(x) + b, \;\; \phi : \mathbf{R} \to F, \boldsymbol{w} \in F$$

In order to minimize the actual risk, the optimized structural risk objective function according to the SRM guidelines is:

$$R_{reg}$$

In order to minimize the actual risk, the optimized structural risk objective function according to the SRM guidelines is:

$$R_{reg} = \frac{1}{2}||w||^2 + R_{emp} = \frac{1}{2}||w||^2 + \frac{C}{n}\sum_{i=1} n|y_i - f(x_i)|$$

$$|y_i - f(x_i)| = \begin{cases} 0, & |y_i - f(x_i)| \leq \epsilon \\ |y_i - f(x_i)| - \epsilon, & |y_i - f(x_i)| > \epsilon \end{cases}$$

Where $||w||^2$ is a description function representing model architecture information, C is a balance coefficient, and $|y_i - f(x_i)|$ is an insensitive loss function of ϵ construction optimization problem solving problem:

$$\min \frac{1}{2}||w||^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*)$$

$$s.t. \begin{cases} y_i - \boldsymbol{w}^T\Phi(x) - b \leq \epsilon + \xi_i^* \\ -y_i + \boldsymbol{w}^T\Phi(x) - b \leq \epsilon + \xi_i \\ \xi_i, \xi_i^* \geq 0, \; i = 1, 2, \cdots, n \end{cases}$$

Where ξ_i^* and ξ_i are relaxation variables, and the interval between $y_i - \boldsymbol{w}^T\Phi(x) - b = \epsilon$ and $-y_i + \boldsymbol{w}^T\Phi(x) - b = \epsilon$ is the regression interval. Lagrange multipliers α, α^* are introduced to transform the quadratic programming problem Dual problem:

$$\max z = \sum_{i=1}^{n} y_i(\alpha_i^* - \alpha_i) - \epsilon\sum_{i=1}^{n}(\alpha_i^* + \alpha_i) \quad (1)$$

$$-\sum_{i=1}^{n}\sum_{j=1}^{n}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)K(x_i, x_j) \quad (2)$$

$$s.t. \sum_{i=1}^{n}(\alpha_i^* - \alpha_i) = 0, 0 \leq \alpha_i, \alpha_j^* \leq C, i = 1, 2, \cdots, n \quad (3)$$

Solving this quadratic programming problem can get the optimal introduction of Lagrange multipliers α_j^*, α_j. When $(\alpha_j^* - \alpha_j)$ is non-zero, the corresponding training sample is to support the forward batch, and the deviation b can be calculated using the KKT condition. Get the expression of the regression function $f(x)$:

$$f(x) = \sum_{i=1}^{n}(\alpha_i^* - \alpha_i)K(x_i, x) + b$$

Where $K(x_i, x) = \Phi(x_i) + \Phi(x)$ is a kernel function that meets the Mercer condition.

Kernel Function

According to pattern recognition theory, linearly inseparable patterns in low-dimensional space may be linearly separable through non-linear mapping to high-dimensional feature space. However, if this technique is directly adopted for classification or regression in high-dimensional space, it is difficult to determine non-linearity. The mapping function forms and parameters, the dimension of the feature space and other inter-problems, and the kernel function technology can effectively solve such inter-problems. There are several types of kernel functions that are commonly used:

- Linear: $K(x_i, x) = x \cdot x_i$
- Radial basis function: $K(x_i, x) = \exp(-\frac{||x - x_i||^2}{\sigma^2})$
- Sigmoid: $K(x_i, x) = \tanh(\kappa(x, x_i) - \delta)$

4 Experiments and Discussion

4.1 Keywords Filter

First, we pick the related word with a Word2vec model. We chose the training model to be the given Word2vec model in a Python package called Gensim. After crawling down 15,000 messages from a future forum, we made our own corpus and put it into the model. After the model is trained, we use the cosine distance as a standard to find out that whether the given two vectors are close enough. Since the (word, vector) pairs are unique, similarity between vectors means similarity between words. We pick 50 most similar related words and build a word list. We intercepted the data of top 20 as shown in the table.1.

Table 1: Similarity for Palm Oil

Keyword	Similarity(cosine-distance)
Soybean Oil	0.982637166976929
Palm Oil futures	0.981469196574831
Open night trading	0.967069029808044
Palm Oil	0.966133223417234
Heilongjiang	0.9574259734153748
Jujube	0.9565946674346924
Corn	0.9463614344596863
Decrease from last trading day	0.9369773292541504
Basis	0.9364948296546936
Agricultural products	0.9264655637741089
funds	0.9163188171386719
Douichi	0.9162120056152344
Request for Comments	0.907367262840271
China Securities Journal	0.9068655705451965
At the meeting	0.9067238903045654
demand	0.8961009621620178
University of Michigan Consumer Confidence Index	0.8931856603622437
Transformation	0.8763970184326172
Backbone	0.8665224862098694
Point	0.8562708353996277

Next, with the related word list, we have to get their Baidu Index. However, in Baidu search engine, not all the related

words exist in the Baidu's word corpus, and have index data. Hence, we first filter out several words that not in Baidu word corpus. (We filter them by typing them into the Baidu Index. If there is no data, we filter this word out; if there is, we keep it).

After the crawling the Baidu Index, those related words have to go through two test layers, the first one is the ADF Unit-Root Test, which can filter out some related keywords that have unstable time series. The second one is the Granger Causality Test, which can filter out some related words that do not have strong causality with the bulk commodity price. As the result of ADF test is shown in table 5, the values are quite clear. Most of the index time series is stable, and all of the DRR and WAR series are stable as well. The Granger causality test is conducted from all the keywords to the volatility of the palm tree commodity price. Table 6 and table.7 reports the p-values of hypothesis testing for Granger causality from GSV to volatility for which the null hypothesis is rejected at 5% significance level. At the end of this step, for each commodity, keywords that do not Granger cause the volatility of price are omitted from the keyword set.

In the end, after the whole procedure, we finally get the related words that may help to predict the model better. The flowchart of keywords filter is as fig.2.

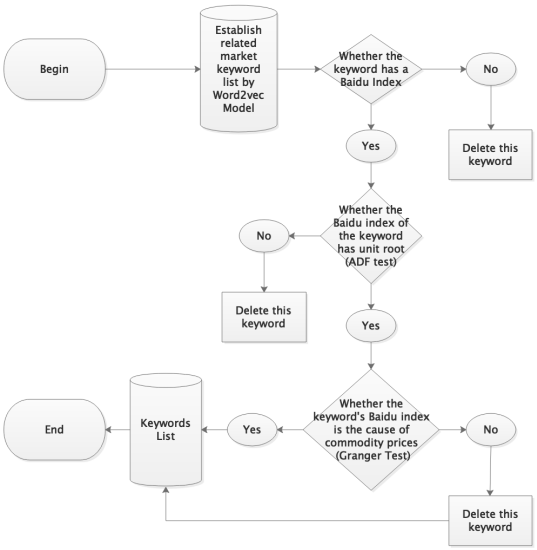

Figure 2: Build keywords list

Taking palm oil futures as an example, we filtered out 50 words related to palm oil through the Word2vec model. By crawling the Baidu index of these keywords, we deleted 26 words. Through the ADF test, we deleted the keywords list to 8 keywords. We performed Granger causality tests on these 8 keywords, and finally selected 3 keywords based on causality.

4.2 Model Validation

In this paper, we divided the data set into a training data set and a test data set with the time node of January 2, 2019. The SVR model calls sklearn's SVM package. Its X feature is the settlement price of the previous three days and the Baidu index of the keywords (the number of days selected based on the lag order), and y is the settlement price trend of the day. I chose accuracy and Root Mean Squared Error (RMSE) as my evaluation indicators because they are widely used in machine learning for classification problems and regression problems respectively. I tested the palm oil, corn, and soybean oil futures with different kernel functions and network attention models. The results are shown in table.2, table.3, and table.4.

Table 2: The Prediction Result of Palm Oil

	Kernel	RMSE	ACC
SVR model w/o index	linear	0.0954	0.6813
	RBF	0.1303	0.5439
	Sigmoid	0.1269	0.0549
SVR model w/ index	linear	0.1018	0.6758
	RBF	0.0955	0.6484
	Sigmoid	0.1043	0.2692
Improvement	linear	-7.13%	
	RBF	26.71%	
	Sigmoid	17.81%	

Table 3: The Prediction Result of Corn

	Kernel	RMSE	ACC
SVR model w/o index	linear	0.2258	0.3901
	RBF	0.2647	0.4615
	Sigmoid	0.2243	0.1978
SVR model w/ index	linear	0.2240	0.4011
	RBF	0.2906	0.4451
	Sigmoid	0.1989	0.1484
Improvement	linear	0.80%	
	RBF	-9.78%	
	Sigmoid	11.32%	

Table 4: The Prediction Result of Soybean Oil

	Kernel	RMSE	ACC
SVR model w/o index	linear	0.1055	0.6593
	RBF	0.1262	0.6538
	Sigmoid	0.1432	0.4231
SVR model w/ index	linear	0.1025	0.6703
	RBF	0.1185	0.6153
	Sigmoid	0.1043	0.4725
Improvement	linear	2.8%	
	RBF	6.10%	
	Sigmoid	27.16%	

The above empirical results found that:

- From the perspective of kernel function selection, all three markets use the sigmoid function as the optimal kernel function, indicating that the sigmoid kernel function can better achieve the non-linear mapping of such series and better stability.

- As for network attention indicators, the three types of market (palm oil, soybean oil, and corn) markets constructed in this paper can capture market participants' attention to the commodity market and related influencing factors in a timely manner, which better represents market participation. Psychological expectations caused by changes in market-related factors. The Granger causality test shows that the Internet attention index can cause commodity prices to some extent. At the same time, the comparison with the benchmark model shows that adding the attention index of the network helps to improve the prediction accuracy of the model.

4.3 Limitations of Research

The limitations of the Granger Company causality test: Granger causality test is susceptible to the choice of time. For instance, if there is a drought in Illinois and Indiana in the summer, then Iowa's corn prices may rise. This pattern, over the last few decades in the US market, is always fit. Still, since about 2008, the United States to promote the use of groundwater irrigation techniques, so after 08 years, the effects of drought and other weather conditions of high temperature and the price of corn in Iowa. At this time, if you use the Granger causality test for the entire period, you will find that the Granger causality test does not match the relationship between Iowa corn prices and drought or temperature. We can tell that the Granger causality test is exceptionally fragile, and several non-conforming coordinates may lead to the failure of the entire test model.

There are certain limitations to the reliability and accuracy of our approach. First, the Baidu Index does not classify the Baidu Index by industry, which may cause search volume in other fields to generate noise in the search data in the financial field that we require. Moreover, Internet search data represents only a fraction of Internet based data that reveal attention. Structuring and analyzing these data requires extremely advanced methods and analyses. We attempt to address this gap by utilizing more accurate employment of search data.

A possible valid concern is that our refining algorithm is kind of a data mining exercise. Being aware of this fact, we are modest in interpreting our findings. In particular, we do not insist that our results demonstrate any causal relationship. Our goal in this paper is to simply improve the predictive power of a conventional volatility pre- diction model (e.g. SVM) by including new sources of information. We begin by a long list of keywords and filter the list by removing keywords with no or little predictive power. It is conceivable that if we used another list of keywords, the final outcome might have been different. However, the fact that the final keywords are quite relevant to the underlying agricutural commodity is to some extent comforting that our exercise is less likely to be a random p-hacking one.

5 Conclusion and Future Work

In this paper, we present that Baidu Index is actually useful for forecasting bulk agricultural commodity price. We present a naive method to find the related keywords with the help of Gensim, and we filter them by conducting ADF Test and Granger Causality Test. Then, with the eminent keyword filtered out, we use a crawler, together with a decrypt function, to catch the Baidu Index for those keywords. Finally, we test our result using SVR model. Moreover, the model with Baidu Index feature performs approximately 10% better.

We still have works to do. Firstly, in Section.3.1, the Word2vec model provided by Gensim is not good enough, especially for its ability to split Chinese words. We may implement our own rules for sentence splitting, adding more stop-words. Secondly, we have the idea that we can make a knowledge graph, which may contains very single agriculture commodity and the keywords relate to them. Then, with the knowledge graph, we may want to figure out the relationship between two different balk agricultural commodities. Are those bulk agricultural commodities with the same related keywords interdependent? Will one bulk agricultural commodity effect another? After figuring out those influence and adding into the knowledge graph, we may better supervise the future market.

References

[Afkhami *et al.*, 2017] Mohamad Afkhami, Lindsey Cormack, and Hamed Ghoddusi. Google search keywords that best predict energy price volatility. *Energy Economics*, 67:17 – 27, 2017.

[Basistha *et al.*, 2015] Arabinda Basistha, Alexander Kurov, and Marketa Halova Wolfe. Forecasting commodity price volatility with internet search activity. 2015.

[Buitinck *et al.*, 2013] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

[Guzman, 2011] Giselle Guzman. Internet search behavior as an economic forecasting tool: The case of inflation expectations. *Journal of economic and social measurement*, 36(3):119–167, 2011.

[Le and Mikolov, 2014] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196, 2014.

[Řehůřek and Sojka, 2010] Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA. http://is.muni.cz/publication/884893/en.

[Ritterman *et al.*, 2009] Joshua Ritterman, Miles Osborne, and Ewan Klein. Using prediction markets and twitter to predict a swine flu pandemic. In *1st international workshop on mining social media*, volume 9, pages 9–17. ac.uk/miles/papers/swine09. pdf (accessed 26 August 2015), 2009.

A

Table 5: Examples of the ADF Test

Name	ADF Result	pvalue	1%	5%	10%
Soybean Oil index	-4.391	0.000	-3.491	-2.888	-2.581
Palm tree index	-4.156	0.000	-3.491	-2.888	-2.581
Coconut Oil index	-3.290	0.000	-3.491	-2.888	-2.581

Table 6: The Granger Causality Test of Palm Oil

Name	F Result	pvalue	lag
Palm Oil Price → Soybean Oil Index	1.0786	0.2991	1
Soybean Oil Index → Palm Oil Price	0.2351	0.6278	1
Palm Oil Price → Palm Index	0.9757	0.4032	3
Palm Index → Palm Oil Price	2.4225	0.0642*[1]	3
Palm Oil Price → Palm Tree Index	2.4277	0.1194	1
Palm Tree Index → Palm Oil Price	4.9949	0.0255**[2]	1
Palm Oil Price → Palm Oil Index	0.6788	0.6788	2
Palm Oil Index → Palm Oil Price	0.3340	0.7161	2
Palm Oil Price → Palm Oil Commodity Index	0.8964	0.4652	4
Palm Oil Commodity Index → Palm Oil Price	0.4767	0.6209	2
Palm Oil Price → Vegetable Oil Index	3.0109	0.0291**	3
Vegetable Oil Index → Palm Oil Price	2.4145	0.1204	1
Palm Oil Price → Rapeseed Oil Index	2.1565	0.0912**	3
Rapeseed Oil Index → Palm Oil Price	0.1160	0.7335	1
Palm Oil Price → Shortening Index	5.7092	0.0170**	1
Shortening Index → Palm Oil Price	3.7327	0.0535*	1

Table 7: The Granger Causality Test of Corn

Name	F Result	pvalue	lag
Wheat Index → Corn Price	6.7276	0.0096***	1
Corn Price → Wheat Index	0.9420	0.3319	1
Paddy Index → Corn Price	1.6807	0.1950	1
Corn Price → Paddy Index	1.0865	0.2974	1
Corn Index → Corn Price	3.6043	0.0578*	1
Corn Price → Corn Index	1.0311	0.3778	3
Corn Origin Index → Corn Price	2.6658	0.1027	1
Corn Price → Corn Origin Index	0.5130	0.5988	2
Corn Price Index → Corn Price	0.1356	0.8732	2
Corn Price → Corn Price Index	0.3055	0.7368	1
Efficacy and Role of Corn Index → Corn Price	2.4753	0.0844**	2
Corn Price → Efficacy and Role of Corn Index	11.6660	0.0000***	2
Nutritional Value of Corn Index → Corn Price	1.4433	0.2364	2
Corn Price → Nutritional Value of Corn Index	1.0044	0.3665	2
Corn Seeds Index → Corn Price	0.4409	0.6435	1
Corn Price → Corn Seeds Index	2.2254	0.1359	1
Corn Market Index → Corn Price	0.2370	0.7891	2
Corn Price → Corn Market Index	0.0628	0.8021	1
Sweet Potato Index → Corn Price	0.6600	0.5767	3
Corn Price → Sweet Potato Index	0.3564	0.7002	2
Peanut Index → Corn Price	5.7092	0.8714	1
Corn Price → Peanut Index	2.2018	0.1109	1

[1]Null hypothesis is rejected at 10% significant value.
[2]Null hypothesis is rejected at 5% significant value.