Business Activity Indicators for Detecting the Impact of Income Information

Luka Kadyntsev[1], Liudmyla Zubyk[1], Serhii Kulibaba[1], Anastasiia Ivanytska[1], and Alona Chorna[2]

[1] Taras Shevchenko National University of Kyiv, 60 Volodymyrska str., Kyiv, 01601, Ukraine
[2] Bogdan Khmelnitsky Melitopol State Pedagogical University, 59 Naukovogo mistechka, Zaporizhzhya, 69000, Ukraine

Abstract

This article presents methods of collecting and analyzing data from media publications, with the intent of establishing relations between them and stock market activity. There exists a great number of solutions, that analyze user-given topics and search for publication activity by given keywords. This solution aims to minimize human input, collecting and categorizing information autonomously, thus minimizing the subjectivity of the result. Currently, the general public's interest in stock market trading is high, but it is near impossible for a human to process all the relevant information and most of the tools are locked behind a paywall or require a certain skill level. The proposed solution is designed to be as simple as possible, while also maintaining a high accuracy in detecting media trends by using clustering and detecting the necessary number of clusters by itself.

Keywords

Algorithm, cluster, media, finance, stock market, newsbreak.

1. Introduction

The stock market is very sensitive to world events [1]. Note that it can be purely economic and political news, or sports, cultural, or news covering some military operations. Thus, by analyzing the news, you can get some valuable information that can help ensure profit during financial operations.

But the task requires pretty heavy automatization. It is impossible to properly process thousands of publications a day, let alone find all the emerging trends in this informational noise. A lot of articles cover unimportant topics, like one-time stories, that would never be picked up by other media outlets.

This leads to the rising need for a system, that would act as a filter to let the user focus on really important news events, that are mentioned in several publications throughout some time. This will greatly reduce the time needed to work through all the news for the day and pick up some missed trends.

In addition, not everyone has the means to perform a large number of natural language processing operations at their disposal. Thus, it would be beneficial for the system to have the ability to be divided into user and server-side applications so that a user can perform data gathering on their machine and then send the data to process on a more powerful machine, that can also be performing other users' tasks.

2. Analysis of Publications, the Status of the Issue, and the Statement of the Problem

2.1. Analysis of Research and Publications

The connection between news and the stock market was established quite a long time ago, but most often research focuses on the

CPITS-2024: Cybersecurity Providing in Information and Telecommunication Systems, February 28, 2024, Kyiv, Ukraine
EMAIL: luca.kadyntsev@knu.ua (L. Kadyntsev); zubyk.liudmyla@knu.ua (L. Zubyk); kulibseryyy@gmail.com (S. Kulibaba); a.titova.wk@gmail.com (A. Ivanytska); chornaa@mdpu.org.ua (A. Chorna)
ORCID: 0009-0002-3864-9318 (L. Kadyntsev); 0000-0002-2087-5379 (L. Zubyk); 0000-0002-7316-1214 (S. Kulibaba); 0000-0002-4803-2090 (A. Ivanytska); 0000-0002-0062-1144 (A. Chorna)

emotional evaluation of news with the subsequent forecast of share price dynamics. For example, researchers from Stanford University, Kari Lee and Ryan Timmons [2], were able to use news analysis to increase the average percentage of profit from trading per month from 0.615% to 2.77%, which is more than a fourfold increase.

Another group of researchers, Dev Shah, Haruna Isa, and Farhana Zulkernin, from Queen’s University [3] was able to achieve an accuracy of 70.59% in short-term prediction of general trends in stock market price dynamics. They also based their research on the overall mood of the publications.

There exist similar works using different algorithms and approaches, but they all have a common problem—they do not handle unsorted data well. All studies note the need to pre-analyze the data for inclusion of non-financial publications. Also, publications consider processing and prediction algorithms but do not consider data collection algorithms, which creates big problems when trying to quickly create large training data sets, having to rely on already existing ones [4–6].

Data collection can be extremely time-consuming and requires a lot of operations. Web scraping can be used, but it can be very slow and a lot of websites have unique page structures and have policies against scraping. Another possible method is reading the RSS feed of a website, but not every website has it.

Processing algorithms also vary greatly, from the simplest word counting to artificial neuron networks with complex preprocessing algorithms [7, 8].

2.2. Analysis of the State of the Issue in the Applied Field

Analyzing data for financial operations has been needed for many years now. While many companies use existing intelligence gathering and processing engines like SemanticForce, they can require a lot of user input and are priced too high for a regular user who is not a corporate entity.

Big financial trading companies often use custom in-house solutions, that focus on some specific markets and are completely inaccessible to the general public.

2.3. Formulation of the Problem

Everyone interested in trading is searching for a competitive edge in information processing tasks. At the same time, existing solutions either do not provide sufficient efficiency or are too expensive and complicated for a potential user.

After the analysis, no solution for the “beginners” sector of trading enthusiasts was found, that would provide adequate service with a good price-to-quality ratio and a low entry threshold.

3. Data Gathering Methods

3.1. Web-scraping

Web scraping is a data collection method in which a website page is loaded and then parsed to extract useful information.

A multitude of tools including whole solutions with customizable scheduling and data formatting exist, with a lot of them being free and open-source.

Among the advantages of this method, the following can be mentioned:

1. Availability of already existing free and open-source software and libraries for web scraping.
2. The information fully corresponds to what the user would see on the page, nothing is lost.
3. The process of web scraping is possible on every site that can be opened in a browser.
4. It is faster than manually collecting data from websites.
5. Results are structured so they are easy to work with.

But this method also has its disadvantages:

1. It is necessary to spend time waiting for the page to load.
2. Sites have different layouts, which forces the scraper to be configured for almost every individual site.
3. Browsers usually “eat up” a lot of operating memory, so a web scraper will not be able to process a lot of pages at the same time.

This method is often used when there is a lot of time and processing power available and websites being scraped have a similar structure. But with a slower machine or unstable internet connection, this method results in being ineffective and slow.

3.2. RSS Feed Reading

Websites use RSS (RDF Site Summary or Really Simple Syndication) to publish information in an XML-like format that can be read with an RSS reader, that is built into most major browsers, and can be installed as a standalone on nearly any platform.

The pros of using RSS are as follows:

1. Very high speed.
2. No need to load the page completely.
3. Sites return only the most important, so there is no need to filter out unnecessary website elements.
4. No need to adapt the reader to each site.

The cons are:

1. Incomplete information—although this filters out unnecessary elements, it can also reduce the amount of necessary information.
2. Some sites simply do not have RSS feeds.
3. Despite the general similarity of the answers, some sites may still differ in one or two fields.

Considering the advantages and disadvantages, the method of RSS requests was chosen. In this case, the speed of forming a massive dataset is more useful than the guaranteed reading of every site. In addition, during data collection, it was found that the percentage of sites without RSS in the Ukrainian news segment is only about 15%, and in the English-language segment—even less.

At the moment, the system can work with sites that have an RSS feed, and it is not necessary to specify the address to access it—if there is a link to the site itself, the system finds the address of the feed automatically. At the same time, in the absence of an RSS feed, the system simply skips the site, displaying a message about the impossibility of receiving information from it.

4. Data Preprocessing

4.1. Tokenization

It was decided to use the spaCy library, which makes it possible to carry out "smart" text tokenization.

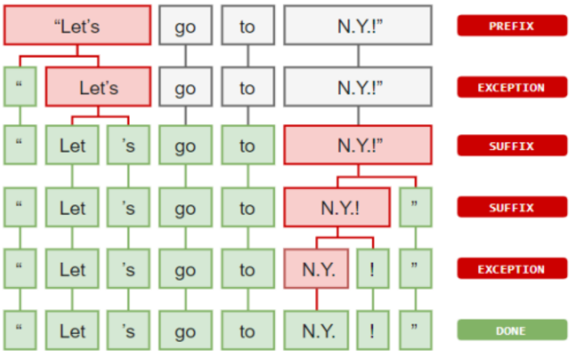

Figure 1: spaCy's smart tokenization

The algorithm can be summarized as follows [9]:

1. Iterate over space-separated substrings.
2. Check whether we have an explicitly defined special case for this substring. If we do, use it.
3. Look for a token match. If there is a match, stop processing and keep this token.
4. Check whether we have an explicitly defined special case for this substring. If we do, use it.
5. Otherwise, try to consume one prefix. If we consumed a prefix, go back to #3, so that the token match and special cases always get priority.
6. If we didn't consume a prefix, try to consume a suffix and then go back to #3.
7. If we can't consume a prefix or a suffix, look for a URL match.
8. If there's no URL match, then look for a special case.
9. Look for "infixes"—stuff like hyphens etc. and split the substring into tokens on all infixes.
10. Once we can't consume any more of the string, handle it as a single token.

Make a final pass over the text to check for special cases that include spaces or that were missed due to the incremental processing of affixes.

4.2. Remove Unnecessary Tokens

During this process, service words are discarded from the total number of tokens, which helps to reduce the amount of information noise.

Registers of stop words for English are much more complete than for Ukrainian. This phenomenon is explained by the "dominant" status of English in programming [10]. However, there is always an opportunity to supplement the lists of Ukrainian stop words yourself, which, although it is an additional waste of time, still significantly improves the final result.

4.3. Lemmatization

Reduction to the original form (or lemmatization) is a powerful mechanism for unifying several forms of the same word. For the processing of the Ukrainian language (because it's a highly inflectional language), this process is necessary [11], since depending on the case, the word can change so much that the result of stemming can be significantly different from the result of stemming the same word in another case.

Lemmatization algorithms work based on existing and trained language models that return the original form of the word by searching their own data sets. As in the case of stop words, very few models support the Ukrainian language at a level sufficient for productive analysis. But, unlike stop words, filling such registers with lems is either very difficult or usually impossible.

Fortunately, the used spaCy library supports the Ukrainian language and has a trained model with the possibility of lemmatization. But even such a trained model is still not able to process the names of cities, leaving them as they are. This is an example of only one of the many problems of processing the Ukrainian language.

5. Clustering

5.1. Hierarchical Clustering

The following scheme can serve as an example of the hierarchical algorithm:

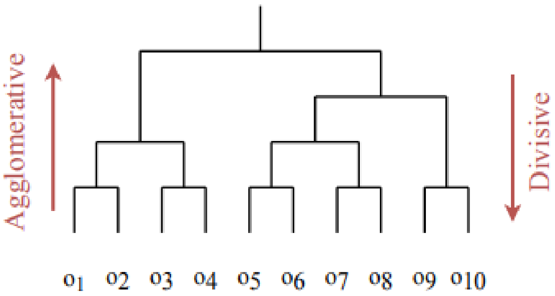

Figure 2: Hierarchical clustering

The hierarchical (agglomerative) clustering algorithm can be described as follows:

1. Designate each point as a separate cluster.
2. Calculate the distance between all clusters.
3. Combine the two nearest clusters into one.
4. If all clusters have not yet been merged into one, then return to step 2.

The distance between clusters can be calculated in several ways.

Single linkage

$$L(r,s)=\min(D(x_{ri},x_{sj})) \quad (1)$$

where r and s are clusters, D() is the distance function, and x_{ri} and x_{sj} are the closest points of the respective clusters.

Complete linkage

$$L(r,s)=\max(D(x_{ri},x_{sj})) \quad (2)$$

where r and s are clusters, D() is the distance function, and x_{ri} and x_{sj} are the furthest points of the respective clusters.

Average linkage

$$L(r,s)=\frac{1}{n_r n_s}\sum_{i=1}^{n_r}\sum_{j=1}^{n_s}D(x_{ri},x_{sj}) \quad (3)$$

where r and s are clusters, D() is the distance function, n_r, and n_s are the number of entries in the corresponding cluster, and x_{ri} and x_{sj} are some points in the respective clusters.

This type of algorithm immediately provides a good visualization of the result, which helps to better imagine the connections between information. Also, it does not require prior information about the number of clusters, which allows the user to independently "trim" the dendrogram at any point. On the other hand, it is precisely this that is a certain minus for this work, because it reduces the degree of automation of the system, clearly forcing the user to make certain decisions regarding the presence of trends, which already brings a certain subjectivity to the process.

5.2. Partitional Clustering

Among the separating algorithms, the k-means and k-medoids algorithms were considered. Algorithms are very similar, but medoids, unlike centroids, are real points, that is, certain entries to the cluster. However, the use of medoids has one undesirable negative side, namely low sensitivity to abnormal points. The specificity of the processed data is such that one cluster often has less than ten entries, with the total number of entries in the thousands, which makes such insensitivity quite critical during processing. Therefore, the k-means algorithm was chosen, which is one of the oldest and most widely used, simple partitioning clustering algorithms.

The algorithm has the following steps:

1. Determine the number of K clusters.
2. Determine K centroids from random points.
3. Assign each point to the nearest centroid, forming a cluster.
4. Calculate dispersion and change the centroid for each cluster.
5. Repeat step 4 until the clusters become stable, the internal variance is minimal, and the variance between clusters is maximal.

5.3. Determine the Number of Clusters

The k-means algorithm requires a certain number of clusters to process. At the same time, the program solves the problem of identifying trends in the news, that is, it must independently determine the emergence of new trends. In addition, it is almost impossible to determine the number of clusters independently on very large data sets. Because of this, there was a need to develop and implement a mechanism for determining the number of clusters.

Table 1
Actual and estimated number of clusters for each algorithm

Algorithm	Estimated n of clusters (actual n = 5)	Estimated n of clusters (actual n = 15)	Estimated n of clusters (actual n = 25)
Elbow	5	10	16
Davies-Bouldin	5	15	23
Silhouette	5	14	25
Calinski-Harabasz	5	15	25
BIC	5	15	25

Multiple algorithms were tested, but the best results were shown by the BIC and the Calinski-Harabasz. But the BIC has one critical flaw—it does not handle well data sets where the number of clusters is not significantly less than the number of occurrences. That's why the solution is using the Calinski-Harabasz index to determine the number of clusters.

The Calinski-Harabasz algorithm can be described as follows:

12. The first step is to calculate the intercluster variance using the formula:

$$BGSS = \sum_{k=1}^{K} n_k \times ||C_k - C||^2 \tag{4}$$

where n_k is the number of entries in cluster k, C_k is the centroid of cluster k, C is the centroid of the entire dataset, and K is the number of clusters.

13. The second step is to calculate the intracluster variance for each cluster using the formula:

$$WGSS_k = \sum_{i=1}^{n_k} ||X_{ik} - C_k||^2 \tag{5}$$

where n_k is the number of entries in cluster k, C_k is the centroid of cluster k, and X_{ik} is the i^{th} entry of the k cluster.

14. After that, you need to add all intracluster variances:

$$WGSS = \sum_{k=1}^{K} WGSS_k \tag{6}$$

where K is the number of clusters and $WGSS_k$ is a measure of intracluster variance for cluster k.

15. Then, calculate the index itself:

$$CH = \frac{\frac{BGSS}{K-1}}{\frac{WGSS}{N-K}} = \frac{BGSS}{WGSS} \times \frac{N-K}{K-1} \tag{7}$$

where K is the number of clusters, N is the total number of occurrences, BGSS is a measure of between-cluster variance, and WGSS is the sum of measures of within-cluster variance. As you can easily guess, it is optimal to increase the intercluster variance, while reducing the intracluster variance, that is, the larger the index, the better.

6. Finding the Trends

In today's world, about 80% of news loses its relevance after 12 hours. Another 10% maintain the level of relevance, and another 10% increase it [12]. Thus, it is the last group that can be called "trends".

In this paper, a trend is considered to be news that is mentioned at least on 2 different days by more than 30% of sources each day. In this way, news that does not hold the attention of readers, or news that was published by several media outlets and published by other media outlets late, but did not receive further development, is filtered out.

By receiving a list of trends, the user can draw appropriate conclusions that can help him make certain financial decisions.

7. Comparison with Analogues

The proposed solution provides users with an easy way to quickly and automatically collect large data samples, determine the number of groups required, and divide the dataset into respective categories. Due to using low-complexity algorithms, the required operational time is minimal and computer resource usage is also quite low [13–30].

While the system is not perfect and does not grant 100% precision, time to effectiveness ratio is satisfactory for a regular trading enthusiast to improve his chances with the financial market.

8. Conclusions

This paper considered algorithms, that are already widely used and proven to be effective.

The principles of algorithms' operations were explained so that users can use custom clustering distance length calculation and choose either lemmatization or stemming.

The proposed system can also be run without an internet connection, by feeding pre-gathered datasets to it, making it possible to use the solution in fully autonomous mode.

As trading is picking in popularity, a decision was made to create an easy-to-use and cheap-to-operate system, that does not require high-end equipment and training. Thanks to this, more people will be able to improve their performance in that field.

References

[1] Y. Ren, F. Liao, Y. Gong, Impact of News on the Trend of Stock Price Change: An Analysis based on the Deep Bidirectiona LSTM Model, Procedia Computer Science 174 (2020) 128–140. doi: 10.1016/j.procs.2020.06.068.

[2] K. Lee, R. Timmons, Predicting the Stock Market with News Articles, CS224N Final Report (2008).

[3] D. Shah, H. Isah, F. Zulkernine, Predicting the Effects of News Sentiments on the Stock Market, IEEE International Conference on Big Data (2018) 4705–4708. doi: 10.1109/BigData.2018. 8621884.

[4] B. Bebeshko, et al., Application of Game Theory, Fuzzy Logic and Neural Networks for Assessing Risks and Forecasting Rates of Digital Currency, J. Theor. Appl. Inf. Technol. 100(24) (2022) 7390–7404.

[5] K. Khorolska, et al., Application of a Convolutional Neural Network with a Module of Elementary Graphic Primitive Classifiers in the Problems of Recognition of Drawing Documentation and Transformation of 2D to 3D Models, J. Theor. Appl. Inf. Technol. 100(24) (2022) 7426–7437.

[6] S. Obushnyi, et al., Ensuring Data Security in the Peer-to-Peer Economic System of the DAO, in: Cybersecurity Providing in Information and Telecommunication Systems II, vol. 3187 (2021) 284–292.

[7] S. Obushnyi, et al., Autonomy of Economic Agents in Peer-to-Peer Systems, in: Cybersecurity Providing in Information and Telecommunication Systems, vol. 3288 (2022) 125–133.

[8] D. Virovets, et al., Ways of Interaction of Autonomous Economic Agents in Decentralized Autonomous Organizations, in: Cybersecurity Providing in Information and Telecommunication Systems, vol. 3421 (2023) 182–190.

[9] Linguistic Features, Tokenization. URL: https://spacy.io/usage/lin-guistic-features#tokenization

[10] U. Abdurakhimovich, The Power of English for Programming. Why is English Important to Software Developers?, Models Methods Increasing Effic. Innov. Res. 3(26) (2023) 145–148.

[11] T. Korenius, et al., Stemming and Lemmatization in the Clustering of Finnish Text Documents, Thirteenth ACM International Conference on Information and Knowledge Management (CIKM '04) (2004) 625–633. doi: 10.1145/1031171.1031285.

[12] C. Castillo, et al., Characterizing the Life Cycle of Online News Stories Using Social Media Reactions, ACM Conference on Computer Supported Cooperative Work, CSCW (2013). doi: 10.1145/2531602. 2531623.

[13] A. Ivanytska, et al., The Advertising Prediction Model Based on Machine Learning Technologies, in: Information Technology and Implementation Vol. 3179 (2021) 35–44.

[14] W. Long, L. Song, Y. Tian, A New Graphic Kernel Method of Stock Price Trend Prediction Based on Financial News Semantic and Structural Similarity, Expert Syst. Appl. 118 (2019) 411–424. doi: 10.1016/j.eswa.2018.10.008.

[15] K. Nam, N. Seong, Financial News-Based Stock Movement Prediction Using Causality Analysis of Influence in the Korean Stock Market, Decis. Support Syst. 117 (2019) 100–112. doi: 10.1016/j.dss.2018.11.004.

[16] Md. E. Karim, S. Ahmed, A Deep Learning-Based Approach for Stock Price Prediction Using Bidirectional Gated Recurrent Unit and Bidirectional Long Short Term Memory Model, 2nd Global Conference for Advancement in Technology (GCAT) (2021). doi: 10.1109/GCAT52182.2021.9587895.

[17] A. Awad, S. Elkaffas, M. Fakhr, Stock Market Prediction Using Deep Reinforcement Learning, Appl. Syst. Innov. 6(6) (2023) 106. doi: 10.3390/asi6060106.

[18] D. Karzanov, Headline-Driven Classification and Local Interpretation for Market Outperformance and Low-Risk Stock Prediction, Computational Econom. (2023). doi: 10.1007/s10614-023-10449-5.

[19] H. Sueno, B. Gerardo, R. Medina, Multiclass Document Classification Using Support Vector Machine (SVM) Based on Improved Naïve Bayes Vectorization Technique, Int. J. Adv. Trends Comput. Sci. Eng. 9(3) (2020) 3937–3944. doi: 10.30534/ijatcse/2020/216932020.

[20] V. Sayoc, et al., Nature Inspired Dimensional Reduction Technique for Fast and Invariant Visual Feature Extraction, Int. J. Adv. Trends Comput. Sci. Eng., 8(3) (2019) 195–200. doi: 10.30534/ijatcse/2019/57832019.

[21] F. Ma'Ruf, Adiwijaya, U. Wisesty, Analysis of the Influence of Minimum Redundancy Maximum Relevance as Dimensionality Reduction Method on Cancer Classification Based on Microarray Data Using Support Vector Machine Classifier, J. Phys. Conf. Ser. 1192(1) (2019). doi: 10.1088/1742-6596/1192/1/012011.

[22] Y. Qiao, M. Alnemari, N. Bagherzadeh, A Two-Stage Efficient 3-d Cnn Framework for Eeg Based Emotion Recognition, IEEE International Conference on Industrial Technology (ICIT) IEEE (2022) 1–8.

[23] Q. Zhang, et al., Transformerbased Attention Network for Stock Movement Prediction, Expert Syst. Appl. 202 (2022).

[24] K. Althelaya, E.-S. El-Alfy, S. Mohammed, Evaluation of Bidirectional Lstm for Short-And Long-Term Stock Market Prediction, 9th International Conference on Information and Communication Systems (ICICS) (2018).

[25] T. Shahi, et al., Stock Price Forecasting with Deep Learning: A Comparative Study, Math. 8(9) (2020).

[26] EOD Historical Data. URL: https://eodhistoricaldata.com/

[27] News API. URL: https://newsapi.org/

[28] Z. Liu, et al., Finbert: A Pre-Trained Financial Language Representation Model for Financial Text Mining, Twenty-Ninth International Joint Conference on Artificial Intelligence (2020).

[29] R. Misra, News Headlines Dataset for Sarcasm Detection (2018). doi: 10.13140/RG.2.2.16182.40004.

[30] F. Xing, et al., Financial Sentiment Analysis: An Investigation into Common Mistakes and Silver Bullets, 28th International Conference on Computational Linguistics (2020) 978–987.