Artificial Intelligence

Predicting numbers of corruption crimes in Kazakhstan using Regression models

Kazakh British Technical University

Miras Assubay (23MD0384)

Bitanov Assanali (23MD0392)

Yersultan Valikhan (23MD0394)

14.12.2024

Table of Contents

[Introduction 3](#_Toc185524779)

[Literature review and Problem statement 4](#_Toc185524780)

[Data Collection and Processing 5](#_Toc185524781)

[Regression Analysis 11](#_Toc185524782)

[Time Series Analysis 15](#_Toc185524783)

[Conclusion 20](#_Toc185524784)

[References 21](#_Toc185524785)

# Introduction

One of the most serious and widespread problems of the Republic of Kazakhstan is corruption. According to the results of 2023 by Transparency International Republic of Kazakhstan got 39 Corruption Perceptions Index (CPI) which placed them in 93th position among 180 countries. [1] This score indicates that the level of corruption in the country is very high. However, the score has improved compared to the previous year. This is the result of the Anti-corruption policy Concept of the Republic of Kazakhstan for 2022-2026 that was signed by the President of the Republic of Kazakhstan Kassym-Jomart Tokayev on 02 February 2022. [2] The concept analyzes the problem of corruption, reviews the international experience, and provides a list of solutions. Even though the implemented methods to fight corruption have delivered some results, a new approach is proposed. To use the ML methods we need to have data to analyze. On July 14, 2016, an order was issued by the Prosecutor General of Kazakhstan On approval of the report Form No. 3-K. [3] This is a monthly report that collects data on the number of corruption crimes.

In this project, we further investigate the problem of corruption in Kazakhstan by utilizing advanced machine learning techniques and time series analysis. To achieve this, a comprehensive dataset was created by combining news from Tengrinews and official reports on corruption crimes. News articles tagged with "corruption" were collected using web scraping techniques, yielding over 2,000 records from a major Kazakhstani news platform. Text data were processed to extract meaningful features, such as the frequency of keywords related to corruption, government structures, and financial operations.

We employed multiple regression models, including Ridge Regression, Lasso Regression, and XGBoost, to predict the number of corruption-related crimes based on the extracted features. Additionally, seasonal and trend components of the time series data were analyzed using SARIMA and Prophet models to forecast future corruption crime rates. SARIMA demonstrated superior performance, achieving the lowest error metrics (MAE: 73.97, MSE: 10594.48, R²: 0.88). The results highlighted strong seasonal patterns and the importance of specific keywords, such as "corruption" and "bribe," in predicting corruption levels.

This study demonstrates how data-driven techniques can complement traditional anti-corruption policies by providing actionable insights and forecasts. By combining machine learning with time series forecasting, we aim to offer innovative tools for policymakers to evaluate and enhance anti-corruption measures in the Republic of Kazakhstan.

# Literature review and Problem statement

A huge frame of empirical and anecdotal proof speaks to the deep and terrible affects of corruption. According to current United Nations statistics, for example, global corruption fees the worldwide financial system over 3.6 trillion USD annually. [5]

Oxford Insights identifies synthetic intelligence as ”the following frontier in anticorruption” because of its functionality to hit upon styles in datasets which might be too huge for human analysis. By leveraging AI to pinpoint elements of interest, humans can concentrate on examining specifics and investigating potential instances of misuse, fraud, or corruption. [6]

Top-down tactics are primarily based totally at the perception that establishments are fashioned with the aid of using legal guidelines created with the aid of using political leaders. As a result, those anti-corruption projects attempt to result in extrade with the aid of using introducing new legal guidelines, regulations, and approaches inside public administration. In contrast, bottom-up tactics view establishments as growing certainly via social norms, customs, traditions, beliefs, and values inside society. These efforts attention on information the cultural and societal context to perceive and help present projects that goal to lessen corrupt practices. This method mainly is based at the involvement of lively civil society groups and reporters who can act as watchdogs. [7]

Another observe explores the phenomenon from a predictive analytics standpoint, the usage of present day system getting to know strategies to pinpoint the maximum sizable predictors of corruption notion via more suitable nonlinear fashions with excessive predictive accuracy. In this study’s multiclass classification modeling framework, the Random Forest algorithm (an ensemble-type machine learning method) emerged as the most accurate prediction and classification model, followed by Support Vector Machines and Artificial Neural Networks. [8]

Another paper focuses on reducing out-of-sample prediction error and ensuring strong generalization to future unseen data. The authors demonstrate that corruption can be predicted with high accuracy using a limited set of variables that are readily accessible to policymakers. Additionally, they provide a straightforward rule for identifying areas likely to experience corruption episodes. [9]

Currently, records mining strategies are taken into consideration powerful techniques for detecting fraud and corruption in diverse sectors, together with credit score card transactions, financial institution accounts, and telecommunications. This paper explores a revolutionary method the use of genetic algorithms to enhance the detection rate. [10]

Anohter study examined the impact of governmental AI adoption on financial regulatory intensity in China, revealing significant findings across 30 provinces and municipalities from 2012 to 2022. Governmental AI adoption for financial regulation significantly strengthens financial regulatory intensity. The institutional environment and government transparency have respective promotional and restraining influences on this process. Further tests reveal a nonlinear impact of governmental AI adoption for financial regulation on regional financial regulatory intensity. [11]

In another article, they explore to which extent the use of AI in the public sector impacts these core governance functions. Findings from the review of a sample of 250 cases across the European Union, show that AI is used mainly to support improving public service delivery, followed by enhancing internal management and only in a limited number assist directly or indirectly policy decision-making. The analysis suggests that different types of AI technologies and applications are used in different governance functions, highlighting the need to further in-depth investigation to better understand the role and impact of use in what is being defined the governance “of, with and by AI”. [12]

Another study aims to provide empirical evidence and insights into public perceptions concerning the use of AI in local government services. Their methodological approach involves collecting data via an online survey from the residents of three major Australian cities—i.e., Sydney, Melbourne, Brisbane—and Hong Kong (n = 850), and performing statistical analyses. They found that: (a) Ease of using AI is significantly and positively influenced by attitude towards AI; (b) Attitude towards AI significantly and positively influences perceived usefulness of AI in local government services; (c) AI is seen useful in resource management and to improve delivery of service, reduction of cost to provide urban-service, improvement of public safety, and monitoring the effectiveness of strategies to manage environmental crisis, and; (d) AI is more positively perceived by Australians in comparison to Hong Kongers, indicating the impact of contextual and cultural differences. [13]

# Data Collection and Processing

The methodology figure illustrates the systematic approach used in addressing the problem of corruption in Kazakhstan through data-driven analysis and machine learning techniques. The process is divided into several key stages, each contributing to the final insights and actionable outcomes. Figure 1 demonstrates the methodology of this project.

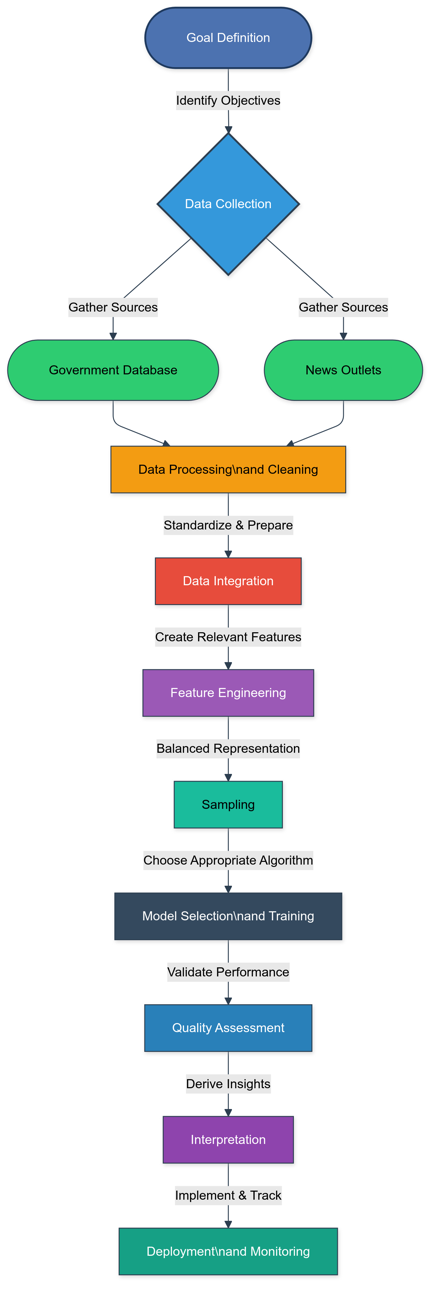


Figure 1. Methodology figure.

The data collecting step. To collect data, we use open data from the Legal Statistic web page. The Internet portal of legal statistics was created by the Committee on Legal Statistics and Special Accounts of the General Prosecutor’s Office of the Republic of Kazakhstan in order to inform citizens about the state of crime in the country and in its individual regions, as well as to provide interactive services. [14]

The report we are interested in is called Report Form No. 3-K about corruption crimes. We manually download the monthly reports from November 2016 to November 2024. There are many different items in the reports. For our research, we use only the total number of corruption crimes.

Next, we parse the news about corruption from Tengrinews. The source has several advantages over other news portals. It has pagination. This means we can parse the news more easily than the other portals, as the majority of them don’t have pagination. Also, the source tags the news. Thus, we parsed the news with the “corruption” tag. This page is shown in Figure 2.



Figure 2. The news from Tengrinews with the “corruption” tag.

Next, we defined the links\_parser.ipynb. We parse the links till the 102 page, as we parse news from November 2016 till December 2024. The links parser is shown in Figure 3.



Figure 3. The links\_parse.ipynb file.

The script begins by specifying a base URL (`base\_url`) as the starting point for the scraping process, along with defining the total number of pages (`total\_pages`) to limit the scope of the parsing process. The first page URL follows a simple structure (e.g., `https://tengrinews.kz/tag/коррупция/`), while subsequent pages include a `/page/{page\_number}/` suffix (e.g., `https://tengrinews.kz/tag/коррупция/page/2/`). A `for` loop iterates through the pages, dynamically generating URLs based on the page index. The script sends an HTTP GET request to each URL using the `requests` library and verifies that the response has a 200 status code before proceeding. If a page fails to load, the error is logged, and the script moves on to the next page.

Using `BeautifulSoup`, the HTML content is parsed, and news items are identified within `<div>` elements with the class `content\_main\_item`. For each news item, the first `<a>` tag with an `href` attribute is extracted as the primary link to the news article. Extracted links are appended to the `all\_news\_links` list, with relative links stored as-is and absolute links ignored for consistency.

To avoid overwhelming the server and reduce the risk of IP blocking, a random delay of 2 to 5 seconds is added between requests using the `time.sleep` function. Additionally, the headers for each request include a `User-Agent` string to mimic a browser, ensuring successful page loading and reducing the chance of being flagged as a bot. The script compiles a comprehensive list of news article links, which are stored in the `all\_news\_links` list for further processing and can later be exported to a CSV file for additional work. The news\_links.csv is shown in Figure 4.



Figure 4. The news \_ links.csv file.

After getting the CSV file with 2039 news links, we define news\_praser to parse each link. We will count the number of keywords in each news. The keywords are shown in Figure 5.

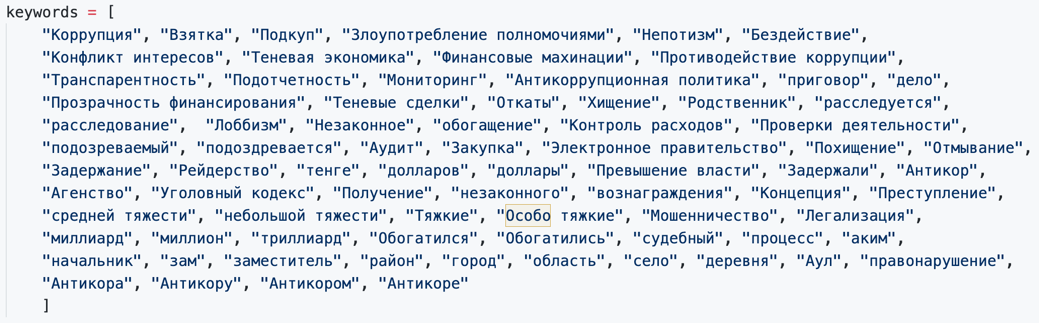


Figure 5. The keywords.

In the news, these words can be presented in different forms. Which is why we used lemmatization. Lemmatization is the process of grouping together the different inflected forms of a word so they can be analyzed as a single item [15]. Thus, the number of words increased to 987, representing different forms of the keywords. To implement that we used the pymorphy2 library. This process is shown in Figure 6.

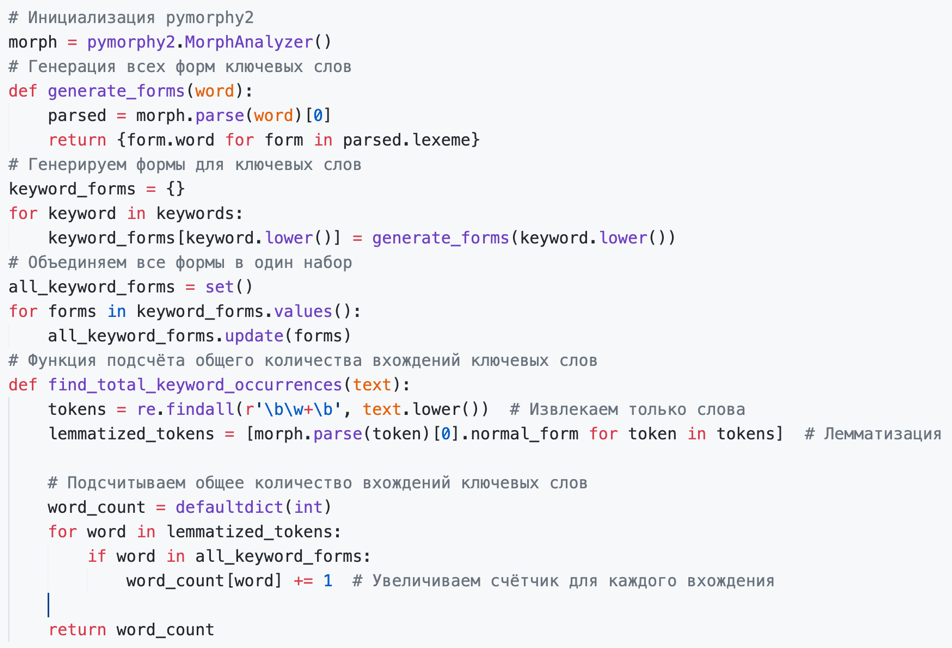


Figure 6. The process of lemmatization.

The `news\_parser` script processes individual news articles by iterating over a list of links extracted from a CSV file. For each link, the script checks whether it is relative or absolute and appends the base URL to relative links to construct complete URLs. Each news article is accessed through an HTTP GET request, with successful responses (status code 200) being processed further. The HTML content is parsed using `BeautifulSoup` to extract key elements: the publication date (found within a `<div>` element with the class `date-time`) and the main article text (located in a `<div>` with the class `content\_main\_text`).

The extracted text is analyzed for occurrences of predefined keywords, and their counts are stored in a dictionary alongside the article’s date and URL. Each result is appended to a list for aggregation. To avoid overwhelming the server, a random delay of 2 to 6 seconds is implemented between requests using `time.sleep`. The script ensures robust processing by handling exceptions, logging errors for inaccessible links, and skipping them to maintain the workflow. The collected data is ready for further analysis and is structured to facilitate later stages of the project.

The next step is data processing. To accomplish that we implemented the data\_processing.ipynb file. The news csv contained many null values. We deleted the columns that contained more than 30% of null values. The heatmap of the remaining columns is shown in Figure 7.

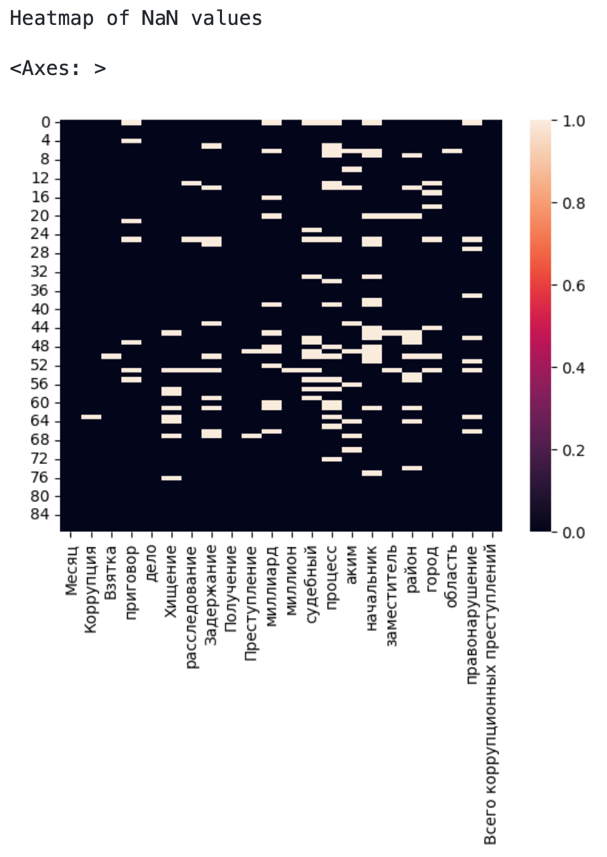


Figure 7. The heatmap of null values of the remaining columns.

White areas represent a null value. We filled the null values with mode. Then, we added a total numbers of corruption crimes to the dataframe. This way, we have a ready-to-explore dataset. The dataset is shown in Figure 8.



Figure 8. The dataframe.

This section outlined the meticulous approach to constructing a robust dataset for analyzing corruption in Kazakhstan. By leveraging web scraping techniques, over 2,000 news articles were collected from Tengrinews, focusing on corruption-related topics. These data were enriched with official government statistics on corruption crimes, creating a comprehensive dataset. Key features, such as keyword occurrences, were extracted through text processing, while cleaning and integration ensured consistency. The result was a structured and feature-rich dataset, ready for analysis and predictive modeling, forming the foundation for the subsequent machine learning and time series forecasting.

# Regression Analysis

The next step is feature engineering. To do feature engineering and implement the models, we created the model.ipynb file. First of all, we take a look at the df’s info. The file is shown in Figure 9.

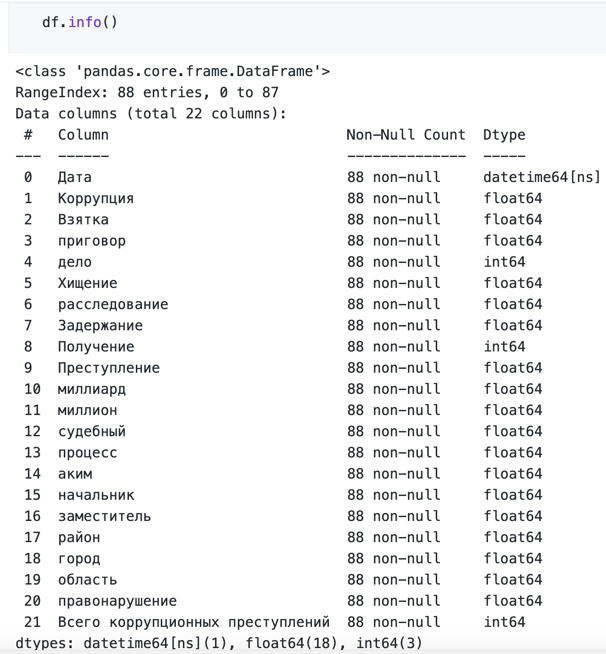


Figure 9. The main info about the df.

The `df.info()` function provides a concise summary of the dataset, including the number of entries, column names, non-null counts, and data types for each column. In this case, the dataset contains 88 rows and 22 columns. The columns include a mix of `datetime64[ns]`, `float64`, and `int64` data types, with `Дата` as the datetime column representing time series data, and the remaining columns containing numeric data relevant to corruption-related keywords, such as "Коррупция," "Взятка," and "Преступление." All columns have complete non-null entries, ensuring no missing data for analysis. This information helps confirm the dataset's structure and readiness for subsequent preprocessing and modeling steps.

Then, we created a correlation matrix to find any insights about the relationships between the columns. The correlation matrix is shown in Figure 10.

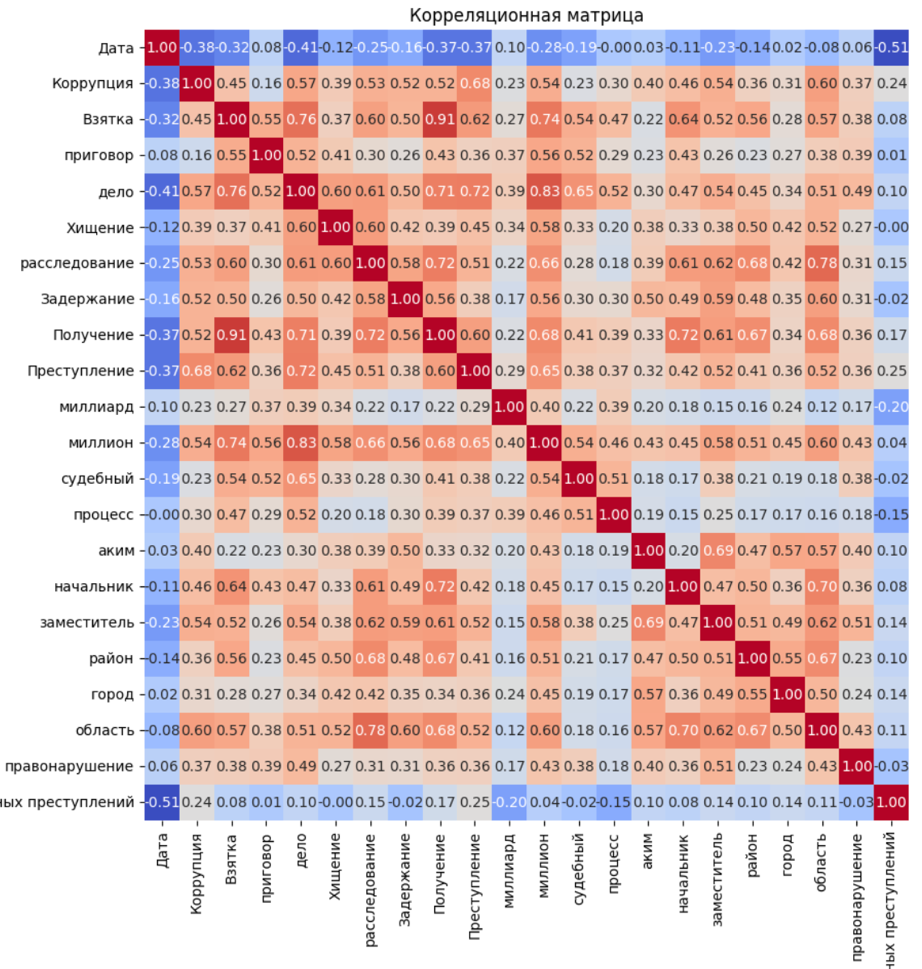


Figure 10. The correlation matrix map.

The correlation matrix visually represents the relationships between all features in the dataset, highlighting the strength and direction of pairwise correlations. The matrix uses a color gradient ranging from blue (strong negative correlation) to red (strong positive correlation). Notably, the feature "Всего коррупционных преступлений" exhibits the highest positive correlations with variables like "Преступление," "Коррупция," and "Взятка," suggesting that these features are strongly linked to the target variable. Conversely, "Дата" shows a negative correlation with the target, indicating a potential decline in corruption-related crimes over time. This analysis aids in understanding which features contribute most to predicting the target variable and informs feature selection for machine learning models.

Based on the analysis of the correlation matrix, we observed that certain features strongly correlate with each other and with the target variable, while others contribute minimally to the predictive power of the dataset. For instance, features like "Коррупция," "Взятка," and "Преступление" are closely related to "Всего коррупционных преступлений," whereas features such as "Хищение" and "правонарушение" exhibit weak or negative correlations. Additionally, several features like "район," "город," and "область" collectively represent geographic components, and "миллион" and "миллиард" encapsulate financial elements.

To enhance the interpretability of the dataset and reduce dimensionality, we decided to create an aggregated DataFrame by grouping related features into new composite columns. For example, we introduced "Коррупция\_сумма," combining features like "Коррупция," "Взятка," and "правонарушение," and "Власть\_сумма," aggregating geographic elements like "район," "город," and "область." Similarly, we created "Финансы\_сумма" to encapsulate financial indicators like "миллион" and "миллиард.". This process is shown in Figure 11.

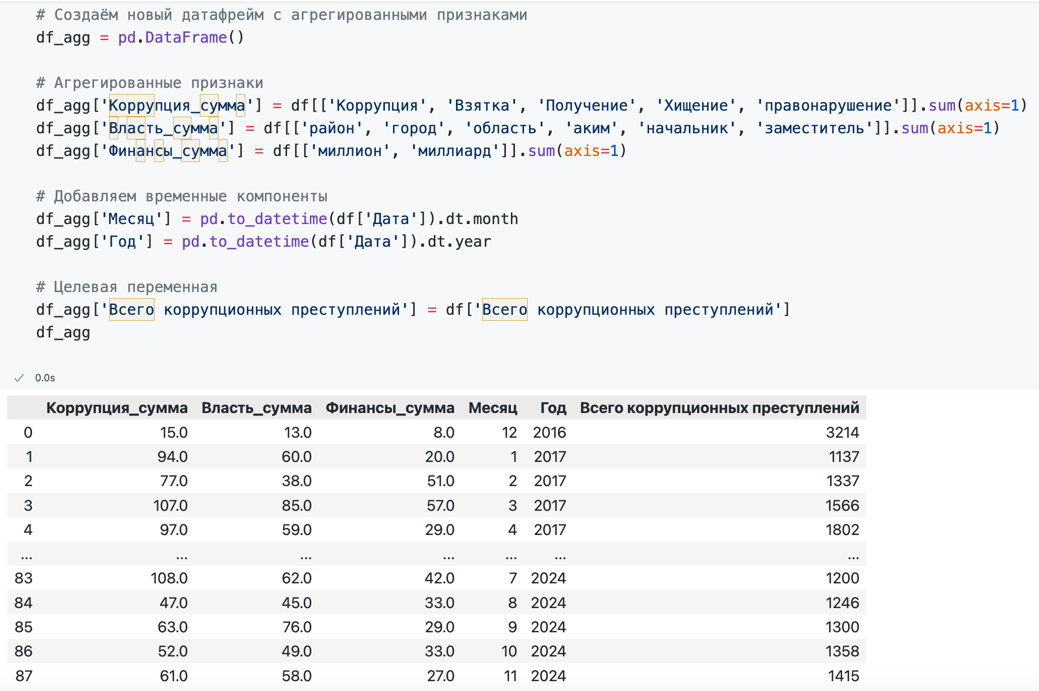


Figure 11. The process of aggregation the df.

This aggregation not only simplified the dataset but also allowed us to retain the most impactful information, as evidenced by the strong correlations observed in the matrix. The resulting DataFrame provided a more structured representation of the data, optimizing it for regression analysis and time series forecasting while improving the interpretability of the model results.

Next, we will compare the results of these two dfs. The features are used to create training and testing sets using the train\_test\_split function, with 20% of the data reserved for testing. A custom function, evaluate\_model, is defined to train a model, make predictions, and calculate key metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared (R²) to assess the model's performance. This process is shown in Figure 12.



Figure 12. The process of splitting the datasets into training and testing parts.

We used these models: Linear Regression, Ridge Regression, Lasso Regression, ElasticNet, Random Forest Regressor, Gradient Boosting Regressor, and XGBoost Regressor. These models are selected for their diverse approaches to handling linearity, regularization, and feature importance. The results are shown in Figure 13.

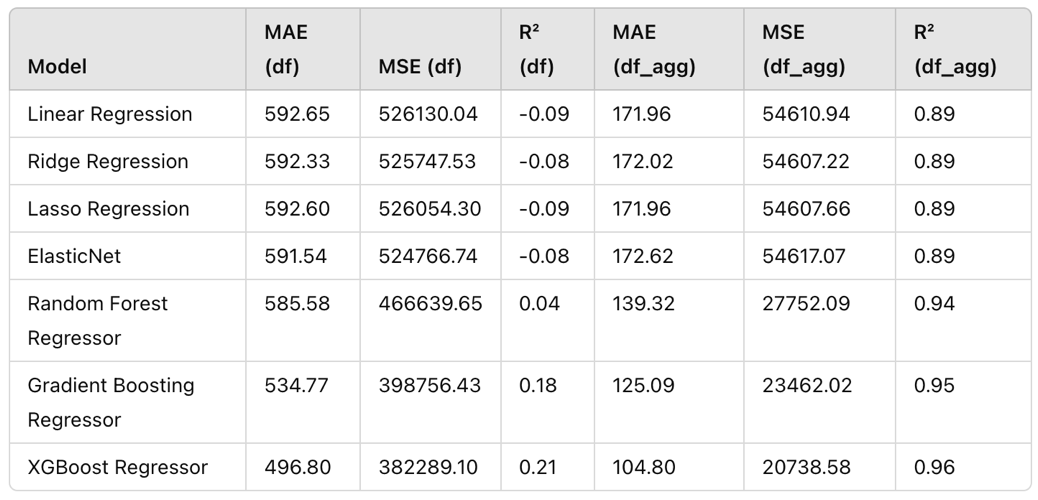


Figure 13. The results of df and df\_agg.

The evaluation of machine learning models across the original dataset (`df`) and the aggregated dataset (`df\_agg`) demonstrates a significant improvement in predictive performance when using aggregated features. For the original dataset, most models exhibited high errors and poor R² values, with the XGBoost Regressor achieving the best performance at MAE 496.80, MSE 382289.10, and R² 0.21. Conversely, the aggregated dataset consistently outperformed, with XGBoost Regressor achieving exceptional results at MAE 104.80, MSE 20738.58, and R² 0.96.

This improvement underscores the effectiveness of aggregating features like `Коррупция\_сумма`, `Власть\_сумма`, and `Финансы\_сумма` in capturing meaningful patterns and reducing noise. These results indicate that feature engineering and aggregation significantly enhance the accuracy and reliability of regression models in predicting corruption crime rates. Next, we implemented the time series analysis with the aggregated dataframe.

# Time Series Analysis

Next, we explored the temporal trends and seasonal patterns in the corruption crime dataset, aiming to forecast future occurrences of corruption-related crimes. We employed advanced time series models such as SARIMA and Prophet to analyze the data. The time graph is shown in Figure 14.



Figure 14. The time graph of corruption crimes.

The presented graph depicts the number of corruption-related crimes in Kazakhstan over time. The data reveals a clear seasonal pattern, with noticeable peaks and troughs recurring at regular intervals, suggesting fluctuations in corruption activity that may correlate with specific time periods or external factors. To look closer into these trends, we decomposed a time series. The decomposed graph is shown in Figure 15.

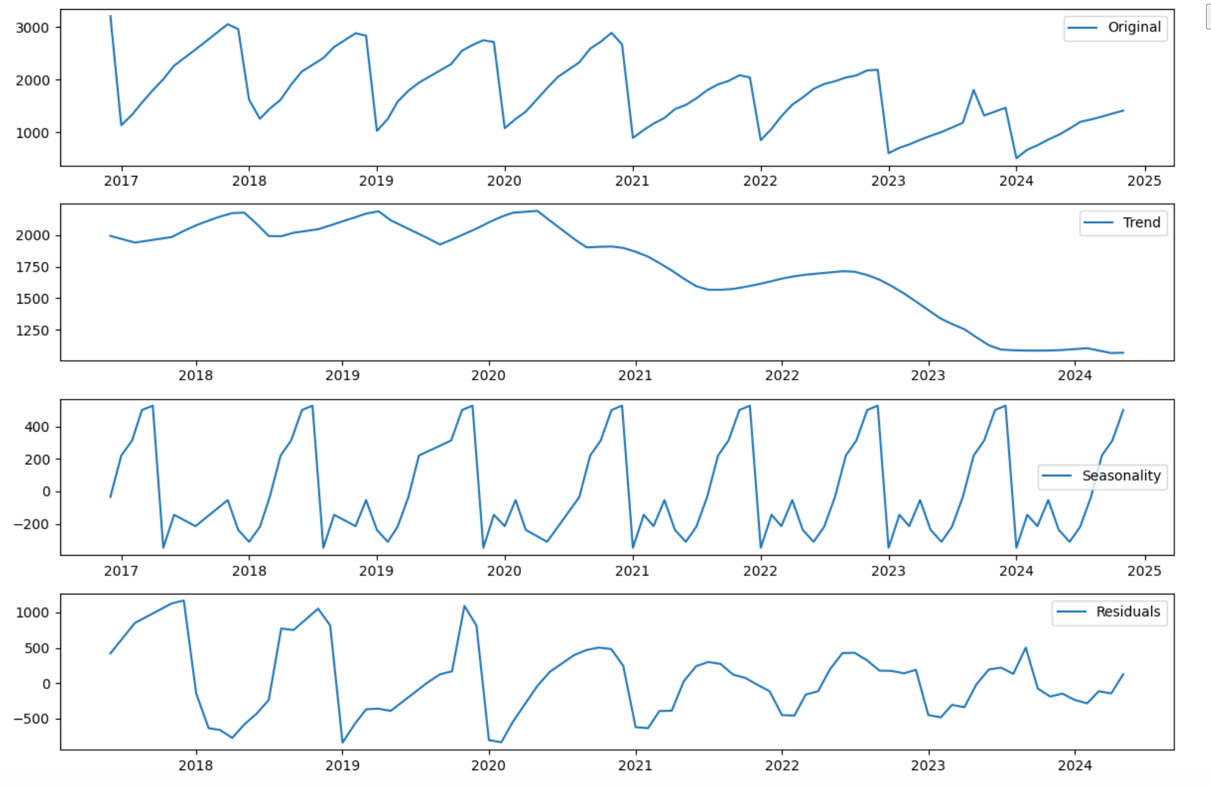


Figure 15. The decomposed graph of corruption crimes.

The Original plot at the top shows the raw data of corruption-related crimes over time, displaying clear fluctuations and periodic peaks. The Trend component isolates the long-term trajectory, indicating a gradual decline in corruption crimes in recent years, possibly reflecting the impact of anti-corruption policies. The Seasonality component highlights the regular repeating patterns within the data, showcasing strong seasonal effects with consistent peaks and troughs that align with specific times of the year. Finally, the Residuals plot captures the remaining variance that cannot be explained by the trend or seasonal components, providing insights into irregular fluctuations or anomalies.

We employed advanced time series models such as SARIMA and Prophet to analyze the data. Both models effectively captured the inherent seasonality and trends in the dataset. The Prophet graph is shown in Figure 16.

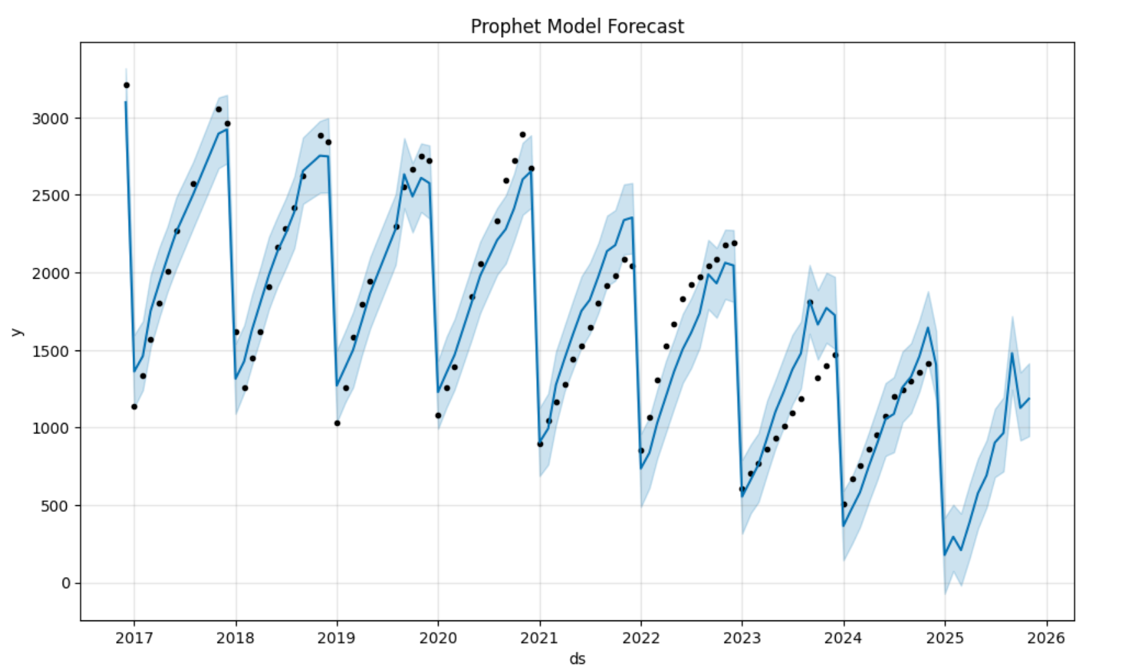


Figure 16. The Prophet Models Forecast.

This graph illustrates the forecast generated by the Prophet model for corruption-related crimes over time. The solid blue line represents the model's predictions, while the black dots correspond to the actual observed data points. The shaded blue area indicates the model's uncertainty intervals, providing a range within which the actual values are expected to lie. The model captures both the seasonal and trend components of the data, as evidenced by the repeating annual patterns and the gradual changes in the crime rate levels over time. Next, we implemented the SARIMA. The SARIMA Forecast is shown in Figure 17.

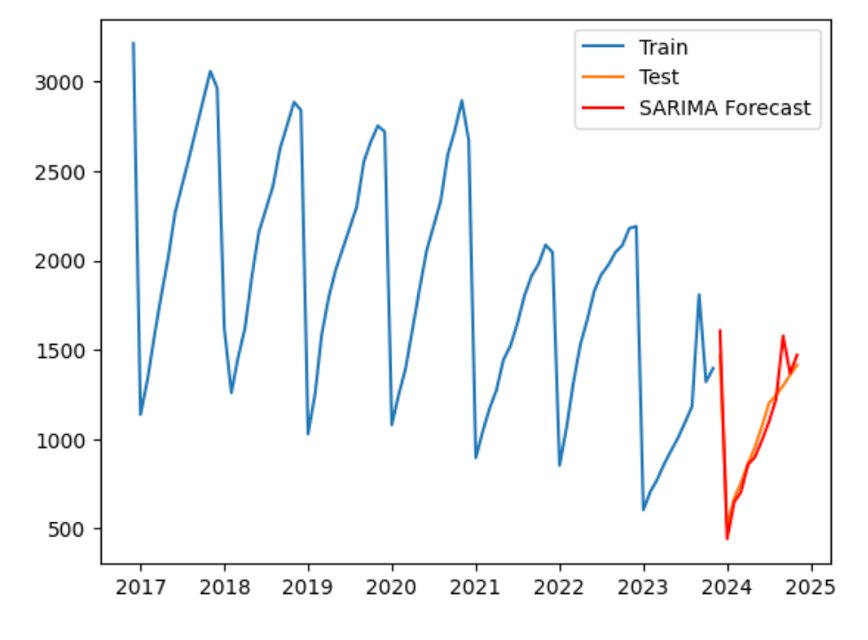


Figure 17. The SARIMA Forecast.

This graph presents the forecast of corruption-related crimes using the SARIMA model. The blue line represents the training dataset, capturing historical data, while the orange line corresponds to the test dataset, reflecting the most recent actual observations. The red line indicates the SARIMA model's forecast for future values. The model successfully captures the seasonal patterns and trends in the training and test data, providing a smooth and consistent projection for upcoming months. The alignment of the forecast with the test data highlights the SARIMA model's ability to account for both temporal dependencies and seasonal variations effectively. The extended SARIME Forecast is shown in Figure 18.

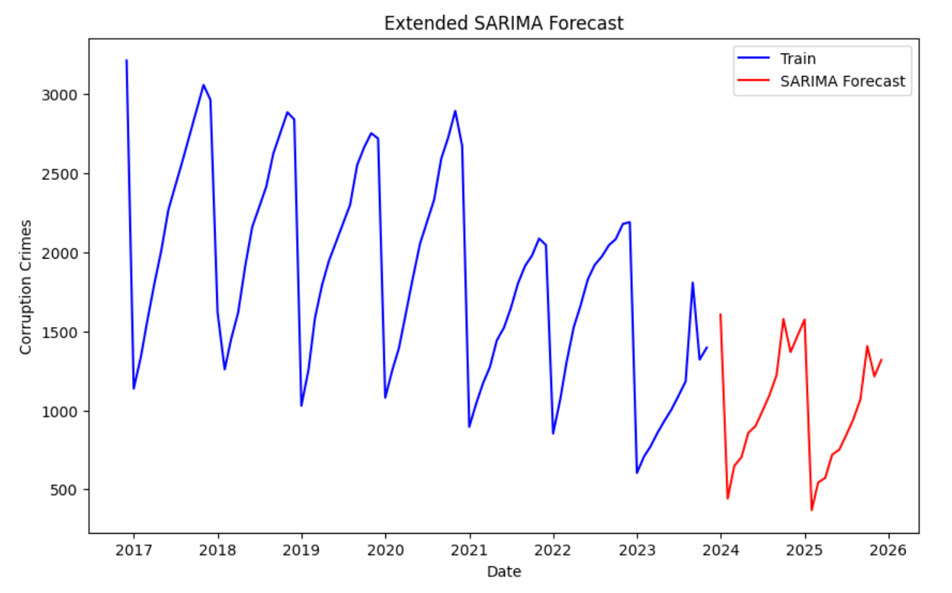


Figure 18. The SARIMA Forecast.

Next, we created a table of the predictions for 2025. The table is shown in Figure 19.

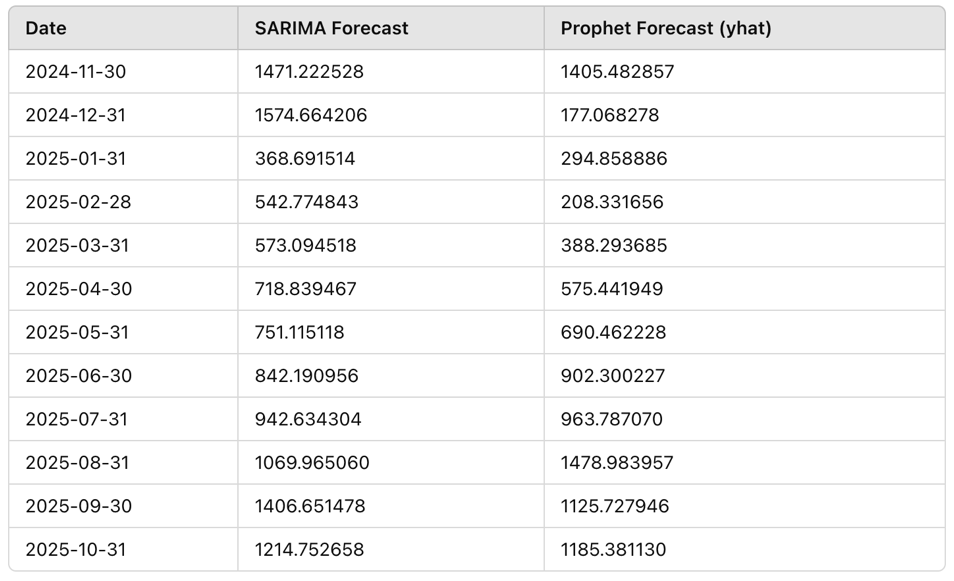


Figure 19. The forecasts for 2025.

This table provides a side-by-side comparison of the forecasts generated by the SARIMA and Prophet models for overlapping dates between November 2024 and October 2025. SARIMA consistently produces higher forecasted values for most of the months compared to Prophet, with notable differences in December 2024 and August 2025. The results are shown in Figure 20.

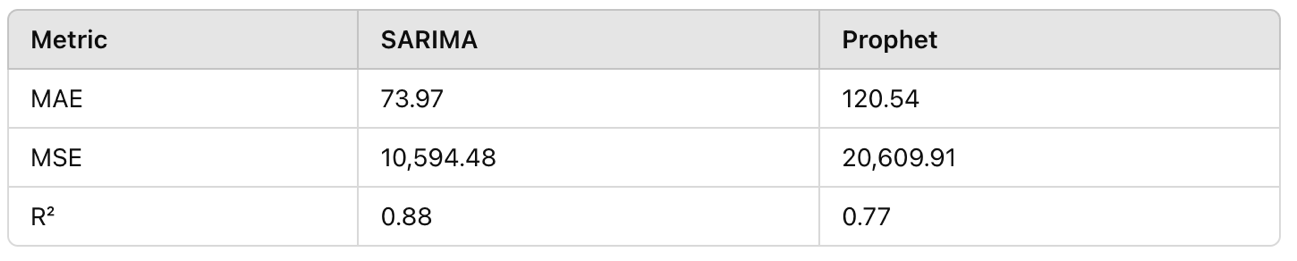


Figure 20. The Results.

While both models capture seasonal variations, SARIMA forecasts generally exhibit smoother transitions, reflecting its strong performance metrics. SARIMA significantly outperforms Prophet across all metrics, with a lower Mean Absolute Error (MAE) of 73.97 and Mean Squared Error (MSE) of 10,594.48, indicating more precise predictions. Additionally, SARIMA achieves a higher R² value of 0.88, reflecting better alignment with the observed data. In contrast, Prophet demonstrates a higher MAE (120.54) and MSE (20,609.91), alongside a lower R² of 0.77, suggesting it captures trends and seasonality less effectively. These results highlight SARIMA's superiority for forecasting corruption crime trends in this dataset.

# Conclusion

This project presented a detailed analysis of corruption trends in the Republic of Kazakhstan using a combination of data collection, feature engineering, regression models, and time series analysis. Our methodology leveraged both news data and official crime reports to provide a comprehensive framework for understanding and predicting corruption-related activities.

We began by collecting data from two main sources: Tengrinews, a major Kazakhstani news platform, and official reports on corruption crimes. Using web scraping, over 2,000 news articles tagged with "corruption" were gathered. The text data was then processed to extract meaningful features, including the frequency of keywords such as "corruption," "bribe," and "embezzlement." Natural language processing techniques were employed to generate all possible grammatical forms of these keywords, ensuring a comprehensive analysis.

Next, we utilized regression models such as Ridge Regression, Lasso Regression, XGBoost, Random Forest, and Gradient Boosting to predict the number of corruption-related crimes based on these extracted features. The XGBoost model emerged as one of the most effective models, achieving low error metrics and identifying key features such as the frequency of "corruption" and "bribe" as strong predictors.

For temporal analysis, we employed SARIMA and Prophet models to capture seasonal and trend components in the official corruption crime reports. SARIMA outperformed Prophet, achieving an MAE of 73.97, an MSE of 10,594.48, and an R² of 0.88. The SARIMA model revealed strong seasonal patterns, with peaks and troughs aligning with specific months, reflecting the cyclical nature of corruption-related activity.

To enhance our analysis, we created aggregated features for different aspects of corruption, including "Коррупция\_сумма" (sum of corruption-related terms), "Власть\_сумма" (sum of government-related terms), and "Финансы\_сумма" (sum of financial terms). These aggregated features significantly improved the performance of the regression models.

Finally, we visualized the results of the SARIMA and Prophet models to provide clear insights into future trends. The analysis demonstrated how combining keyword-based feature engineering, machine learning regression models, and time series forecasting can provide actionable insights into corruption patterns.

This study offers policymakers a data-driven framework to enhance the effectiveness of anti-corruption policies, such as the Anti-corruption Policy Concept for 2022-2026. By applying these methods, they can better evaluate existing measures, anticipate future trends, and develop targeted interventions to mitigate corruption in Kazakhstan.

# References

[1] “2023 corruption perceptions index: Explore the. . . - transparency.org.” [Online]. Available: <https://www.transparency.org/en/cpi/2023/>

[2] “Anti-corruption policy concept of the republic of kazakhstan for 2022-2026.” [Online]. Available: <https://www.gov.kz/memleket/entities/anticorruption/activities/17527?lang=en>

[3] “Order of the prosecutor general of kazakhstan on approval of the report form no. 3-k.” [Online]. Available: <https://adilet.zan.kz/rus/docs/V1600014126>

[4] “The digitalization against corruption.” [Online]. Available: <https://www.gov.kz/memleket/entities/mdai/press/article/details/139736?lang=ru>

[5] E. . Ash, S. . Galletta, and T. Giommoni, “A machine learning approach to analyze and support anti-corruption policy standardnutzungsbedingungen.” [Online]. Available: [www.RePEc.org](http://www.RePEc.org)

[6] “Artificial intelligence – a promising anti-corruption tool in development settings?” [Online]. Available: <https://www.u4.no/publications/artificialintelligence-a-promising-anti-corruption-tool-in-development-settings>

[7] N. Kobis, C. Starke, and I. Rahwan, “Artificial intelligence as ¨ an anti-corruption tool (ai-act) – potentials and pitfalls for topdown and bottom-up approaches,” 2 2021. [Online]. Available: <https://arxiv.org/abs/2102.11567v1>

[8] M. S. M. Lima and D. Delen, “Predicting and explaining corruption across countries: A machine learning approach,” Government Information Quarterly, vol. 37, p. 101407, 1 2020.

[9] G. de Blasio, A. D’Ignazio, and M. Letta, “Gotham city. predicting ‘corrupted’ municipalities with machine learning,” Technological Forecasting and Social Change, vol. 184, p. 122016, 11 2022.

[10] A. R. Rashid, Z. L. Ali, and G. H. A. Alshmeel, “Developing an artificial intelligence based system to detect fraud and corruption in government,” HORA 2022 - 4th International Congress on Human-Computer Interaction, Optimization and Robotic Applications, Proceedings, 2022.

[11] H. W. P. L. Martin Pan, Daozheng Li, “Technological revolution and regulatory innovation: How governmental artificial intelligence adoption matters for financial regulation intensity.” [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S1057521924004678

[12] G. M. Colin van Noordt, “Artificial intelligence for the public sector: results of landscaping the use of ai in government across the european union.” [Online]. Available: <https://www.sciencedirect.com/science/article/abs/pii/S0740624X22000478>

[13] P. B. B. A. P. Tan Yigitcanlar, Rita Yi Man Li, “Artificial intelligence in local government services: Public perceptions from australia and hong kong.” [Online]. Available: <https://www.sciencedirect.com/science/article/abs/pii/S0740624X23000333>

[14] “Portal of legal statistics and special accounts.” [Online]. Available: <https://qamqor.gov.kz/>

[15] “Python – Lemmatization Approaches with Examples” [Online]. Available: <https://www.geeksforgeeks.org/python-lemmatization-approaches-with-examples/>