# OSINT Visualisation

1st Ollie Myers
*Computer Science, MEng*
*University of Bristol*
Bristol, United Kingdom
al20790@bristol.ac.uk

2nd Luke Sakaguchi-Mawer
*Computer Science, MEng*
*University of Bristol*
Bristol, United Kingdom
gq20332@bristol.ac.uk

3rd Radmehr Ghassabtabarshiadeh
*Computer Science, MEng*
*University of Bristol*
Bristol, United Kingdom
we20153@bristol.ac.uk

4th Hanbin Zhang
*Computer Science, MEng*
*University of Bristol*
Bristol, United Kingdom
ey20699@bristol.ac.uk

5th Pablo Sanz Maroto
*Engineering Mathematics,MEng*
*University of Bristol*
Bristol, United Kingdom
gb20778@bristol.ac.uk

6th Ashiph Rai
*Engineering Mathematics, MEng*
*University of Bristol*
Bristol, United Kingdom
lo20958@bristol.ac.uk

## I. Abstract

Military expenditure accounts for a large proportion of a country's total spending, with weaponry travelling across the world in order to power brutal conflicts. We wanted to create a simple tool that would help us visualise and understand the transfer of arms over the past 70 years. We also wanted to predict any potential trends in military expenditure, such as the expected forecast, as well as which countries tended to trade with each other.

## II. Introduction

Open Source Intelligence (OS-INT) [1] is defined as the act of extracting and analysing information from publicly accessible sources. Its purposes include, but are not limited to national security, military intrigue, business intelligence and academic research [2]. With the rapid growth of the reach of the internet, OS-INT has reached new highs in levels of practitioners, as well as the breadth of the content illustrated by the data.

For this project, we are working with several different sources of data, and OS-INT practitioners, such as the Stockholm International Peace Research Institute (SIPRI), who provide data which encapsulates military expenditure, armed conflict and the arms trade.

With this project, we hope to identify and visualise the direct correlation between the transfer of arms and equipment to the levels of conflict across the globe.

## III. Related Work

Statistically forecasting any sort of government expenditure, let alone military, is a process done by many academics in order to develop an insight as to how a country's tax is being invested. Previous studies have been done on the two most populous countries, China and India. The models presented show strong predictive power over previously used models, such as artificial neural networks. One study which predicted India's military expenditure [3] showed that ARIMA models could predict military expenditure with an accuracy of at least 95% if tuned correctly.

## IV. Project Goals

Before starting work on the project, we initially identified our goals. We expressed interest into using different machine learning models to find trends in the data, however we were originally unsure about how best to approach visualising our findings,as well as the dataset as a whole.

For the identification of trends aspect of this project, we decided to utilise numerous machine learning methods. Initial research into visualising time series data was mainly focused on using different charts to display the data. Although this could be used to display some of the findings of our machine learning methods, we wanted something that allowed someone without technical understanding to be able to explore the data visually. Further research led us to discover Kapersky's cyber attack map [4]. This site displays real-time data on cyber attacks. We agreed that this format was much better for our use case,however we wanted to make our version simpler so that anyone could use it. Therefore we opted to create a website that allows users to pick a date and have the arms trade for the year be displayed in an easy-to-understand format.

## V. Data Preparation

Before we could start applying machine learning methods, or creating our visualisation website, we first needed to acquire and clean our data. Polars was used instead of conventional pandas dataframe. Polars is believed to be faster and more robust and provided an overall smoother experience. [16]

The first major dataset we looked at was the SIPRI TIV (Trade Indicator Value) dataset [5]. The data contained in this dataset is on all arms trades from 1950 to 2022 (as of the time of writing).

Access to this data is public but the page to download this data was only provided through a contact at SIPRI as this download page is hidden. In spite of this, the data is still being updated as there are seemingly other ways to access this data from other pages.

Due to this site not being well maintained, there was a major issue that had to be dealt with. Unfortunately, requesting multiple years at a time would cause the site to drop the connection. For the sake of making the application future proof, the site was checked to get a list of available dates and if the currently held data was out of date, we would request the Comma Separated Value (CSV) file of each year one at a time. The piece of code that handles this be run as a cron job or using Windows task scheduler to ensure that the data is consistently kept up-to-date. After all the data is gathered it is then be combined into a single CSV which is then used in the rest of the project.

This dataset comprises approximately 58,000 records spanning from 1950 to 2022. The first six lines of the CSV file provide essential information about the dataset, including the name of the dataset on the first line and the year range on the fourth line. The sixth line contains the column names of the dataset. The original files obtained from the SIPRI website include an additional blank line and a line of clarification at the end of the file, which are not present in the combined file. Each register of the dataset contains the following values:

1) Deal ID, which is a unique ID for each deal.
2) Seller and Buyer countries.
3) Designation, Description, and Armament category of the weapons.
4) Order date and Delivery, Numbers delivered, Delivery year
5) Indicator for the is the values in the previous item being estimated
6) TIV deal unit and TIV delivery values
7) SIPRI estimate and Local production

It is important to note that a single arms deal may span over multiple years, resulting in multiple records with the same Deal ID. This indicates that these records belong to the same deal, and should be considered as a whole when analyzing the data. Additionally, the dataset includes the TIV (trend-indicator value) system developed by SIPRI [6] to measure the volume of international transfers of major conventional weapons using a common unit. This metric reflects the military capability of the weapons involved in the deal and is a valuable feature for analyzing the dataset.

The second of the SIPRI datasets we used was their trade register dataset [7]. This dataset contains more detail regarding trades between countries. We used this to validate the TIV dataset and visa versa. The main issue discovered with the data gathered, was the format in which SIPRI's site provided access. The download page gave an RTF file which was incompatible with the CSVs from the TIV dataset. In order for the two sets to be used in tandem, RTF files had to be converted into CSV. A custom script for this was written to ensure that all required data was converted into a CSV that could then easily be used to verify the TIV data.

In order to ensure that application we are building is future proof, we again developed a script in python that would periodically request new RTF files for the transfers in order to ensure we had the most up-to-date version of the data hosted by SIPRI. Instead of using a web-scraper that would manually change the start and end date of the data we were interested in, we created a simple script that just sends a HTTP request to download the data.

This dataset contains approximately 28,000 transfer registers, which is less than half the number of registers in the SIPRI TIV dataset. This difference is due to the fact that this dataset treats transfer registers with a range of delivery years as a single register, whereas the SIPRI TIV dataset separates them into multiple registers. Unfortunately, this data didn't contain the 2022 data (as of the running of all of our analysis), so we were unable to use this to validate the 2022 TIV dataset.
Each register of the dataset contains the following values:

1) Supplier and Recipient countries.
2) Ordered and Of Delivered, the ordered number and the actual delivered number.
3) No. Designation and Weapon Description.
4) Order date and Delivery, Numbers delivered, Delivery year
5) Year(s) Weapon of Order and Year Delivery.
6) No. Comments

The Year Delivery in the dataset actually represents a range of years, whereas the Year(s) Weapon of Order does not contain a range of years. This was confirmed by checking for the presence of a hyphen ("-"), which is commonly used to indicate a range in this dataset. Additionally, the dataset uses brackets to indicate whether a number is estimated by enclosing the estimated values or times in brackets.

The combination of this data also presented a number of issues. The first dataset contains most of the information in the second dataset, except for the "No. comments" column, which provides a brief introduction for the arms transfer. Therefore, dataset one was chosen as the primary dataset for visualization and analysis. However, the "No. comments" column from the second dataset was joined to dataset one to provide additional information. To ensure consistency between the two datasets, several processing steps were taken. First, each value in the "Year(s) Weapon of Order" column of the second dataset was checked and any brackets were removed. Additionally, some irregular characters in RTF files were found to be encoded in a default unreadable format and displayed as their unicode codes. To solve this problem, we employ regular expressions to identify patterns of those irregular characters and convert them into readable unicode format. Finally, a left join was performed, with the matching values left on Seller, Buyer, Designation, and Order date in dataset one, and right on Supplier, Recipient, No. Designation, and Year(s) Weapon of Order in the second dataset. This allowed for the "No. comments" column to be added to dataset one while maintaining consistency between the two datasets.

At this point, we started to look into how we would build the website that could display a high level overview of this data. Given we decided on showing this through a map, we would need to have a reference to locations of the countries, as countries border have changed a lot since 1950. Research into how to access this kind of data was initially unsuccessful, as there are no direct ways to query for a specific countries location during a specific year. Despite this setback, we kept looking for ways to access this kind of data and eventually discovered an online atlas website, Ostellus [8].

We thought that this site may have some kind of API that it uses to poll it data, for displaying on the front end. Fortunately this assumption was correct, and we discovered their hidden and undocumented API. By observing the network requests, we were able to get a base understanding of how to get the exact data we were looking for. The API required a seemingly unrelated id-string to get the data for each country. The range of these ids were manually found and a script to pull out all of this data was created. Due to the way website is built, sending too many requests would cause the download to time out and fail, so the approximately 3400 ids had to be sent in batches of 10 with a 10 second wait between each batch. This was extremely inefficient, and the data required a lot of manipulation before it could even be used.

Looking for a way to future proof this, we quickly realised that our manually found range was only a subset of the overall data, and that there was no quick or efficient way to find this range. We then turned back to Ostellus site to attempt to find if they were loading their data in a different way. this involved digging through a client-side JavaScript file that was over 100,000 lines long to try and find a single function call that grabs the data. This digging caused us to discover a call to a single JSON file that contained all of the information we needed on countries' borders. The downloading of this JSON document is its own script and so like the other two datasets, can be scheduled using cron or Windows task scheduler.

Another dataset that is utilised for analysis was the global expenditures dataset. This dataset outlines various different ways to measure expenditures from total spending in US dollars to percentage share of GDP. After inspecting each measure, we decided to use the constant version of total US dollars spent. This measure, much like the other measures, has a yearly report of how much money was spent in US dollars each year for documented 174 countries. The constant US dollar was chosen as it is easier to capture relationships between a value that is scaled consistently throughout the data. If inflation was taken into account, due to the fluctuations in the value of the US dollars, capturing a linear relationship may have been more difficult. To clean this data, standard procedures were followed and did not require excess data manipulation. This involved removing empty or unnecessary rows and replacing NaN values with an appropriate substitute - in our case we chose -1. Since NaN values are attributed when there is no available data, usually because the country no longer exists, having the value as -1 makes sure that it is not mistaken for zero spending as they have two different meaning.

Since SIPRI's goal is to find a solution to a more peaceful world free from conflicts, we decided that an additional dataset should involve conflicts. This way, we can identify whether conflicts are influenced by Government spending on the military or arm trades deals. After, researching into finding a related dataset, Uppsala Conflict Data Program (UCDP) [9] was selected. This dataset encompassed the key elements that was needed for the analysis - this includes using a similar pool of countries and covering a closely matched time interval. To summarise the contents of the data, the data describes every frequency of a conflict from 1946 to 2021 - in total there is 2568 occurrences of conflicts within the dataset. Key features of the data include location of conflicts and countries involved in the conflict. To translate this data into a similar form as the global expenditures dataset, a script was compiled to iterate through occurrences of conflicts and count the frequency of involvements for each country. By the end of this process, the dataset is transformed to exclusively include columns for countries and a column for total number of involvements in conflicts.

Overall, the conflicts dataset included 119 unique countries and the global expenditures included 174 countries. After merging the two datsets together, the total expected number of countries is 119 - the minimum number of countries between the two dataset. Merging the two datasets resulted with data for 77 countries. Since 35% of the data is lost from the merging process, providing analysis on the available subset of the data can be misleading and inaccurate. This issue was caused by the procedure used to label the countries. A large proportion of countries that were lost in the merging process was due to countries having a 'second label'. For example, Myanmar was labelled as 'Burma (Myanmar)'. This is due to the fact that country names have changed since the first instance of conflict within the dataset. This differs with the global expenditures dataset where all countries are labeled based on 2021's country names. To solve this issue, regular expressions were constructed to extract and replace the country labels where there were occurrences of bracketed country names. Even after this process, we were left with 100 countries. The remaining countries that all have specific issues in the naming. For instance the Soviet union is 'soviet_union' in the conflicts dataset and 'USSR' in the global expenditures dataset. Due to these issues all having different naming procedures, this problem could not be swiftly programmed into a common name and will require changes to be made manually.

## VI. Data Exploration

Before we started looking into using machine learning or graphing techniques on our combined data, we first did some basic exploration of our dataset. This involved creating some high level graphs so that we could start to learn what our data looked like and to get an overview of some important statistics.

One of the key statistics we chose to look at in this exploration stage was how the number of weapon orders looks like over time.

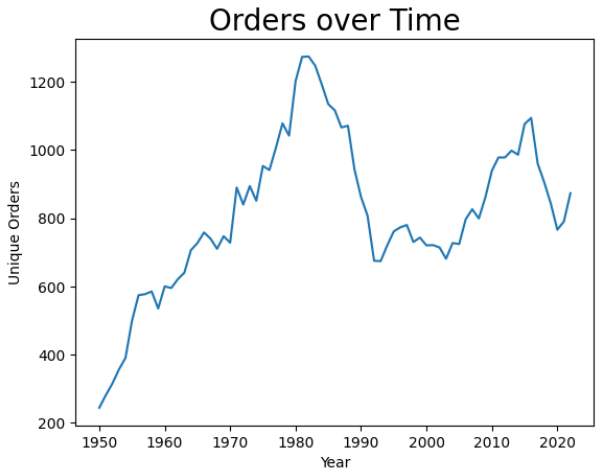



Fig. 1. Number of orders plotted against year

The graph shows the clear hot spots of activity during the turn of the decade of the 1980's and the late 2010's, specifically 1982 and 2016. During those years different conflicts have been happening around the globe such as Iran-Iraq wars, the invasion of the Falkland islands and different civil wars in Middle East on 2016. Using the arms trade dataset available, trades are showcased on the map and there is a clear spike in the volume of trades. The biggest supplier is United States. Argentina, who invaded Falkland islands, bought most of their weapons that year from Peru, Italy and United States. An interesting observation is the fact that Italy is huge supplier to conflict ridden countries in the world. Italy has bought most their supply of weaponry from United States and United Kingdom in the years before 1982.

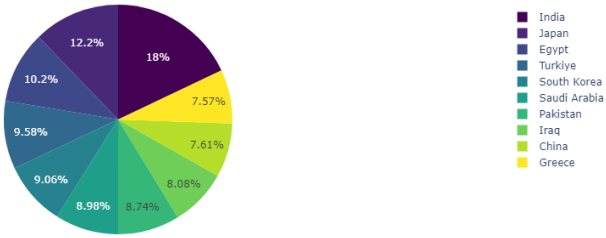



Fig. 2. Top 10 Buyers

The graph above shows the top ten countries based on number of orders over time. As seen, the countries listed have either had conflicts in recent history (Egypt, Iraq, Pakistan), or are currently in a state of rising tensions (Japan v North Korea(not pictured) [10]), India and China [11]), hence it makes sense that these countries would begin stockpiling weapons.

In this graph, the top ten sellers of weaponry are shown, which clearly relates to the list of the top ten weapons manufacturers [12], with US-based companies holding the top 5 places, UK-based BAE-Systems coming in at 6th, and the rest being located in China.

Figure 4 shows the total military spending for each country

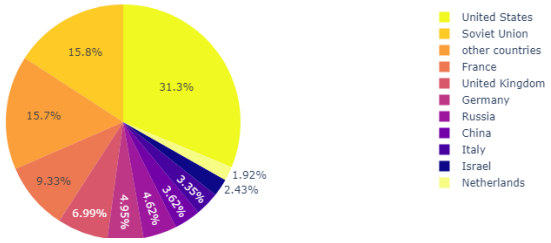



Fig. 3. Top 10 Sellers

every year and visualises the trend of this particular feature. This graph creates a general understanding of how spending towards the military has changed since 1949 to 2021.

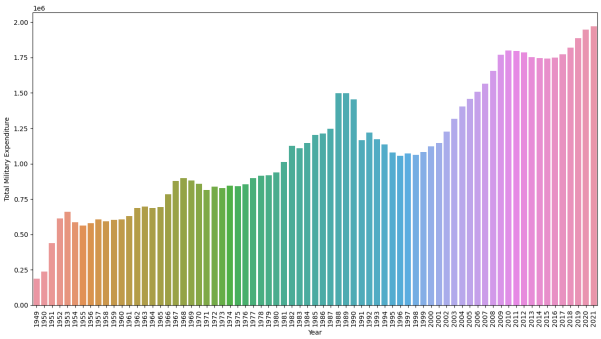



Fig. 4. Total military expenditures per year for all countries - SIPRI Global Expenditures dataset

Much like figure 4, the plot from figure 5 shows graph for the total number of conflicts for each year. The comparison between figures 4 and 5 provides interesting insights as both trends resemble each other closely. As a result, we decided to note this down and decided to run machine learning analysis with hopes in finding a relationship between the two features.

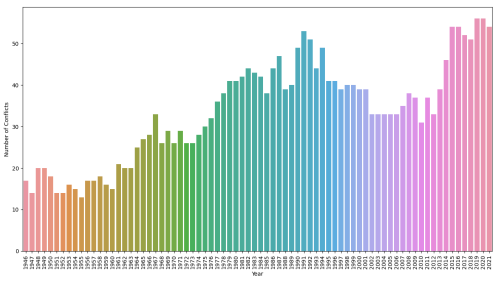



Fig. 5. Total conflicts per year for all countries - UCDP Conflicts Dataset

## VII. Data Modelling

### A. Linear regression

*1) Linear regression to predict future expenditure:* An initial statistical method which we decided to employ was to

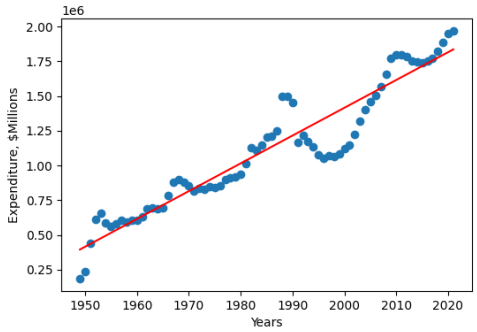



Fig. 6. Expenditure over time

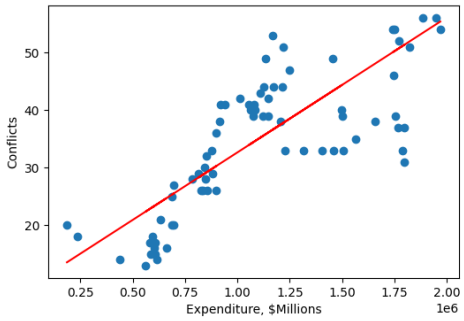



Fig. 7. Expenditure plotted against total conflict

determine the Military expenditure of the world as a whole, based on the year. Utilising the dataset for military expenditure from SIPRI which helpfully had a page which stated each country's expenditure, we were able to plot expenditure against time. As seen, there is a clear pattern of increasing military expenditure over time.

When evaluating the model, we were presented with large values of error, so in order to analyse the error in a more reasonable manner we applied the logarithm function to the expenditure values, given that they were so massive. After this, we were able to assess the mean squared error of the training dataset at an average of 0.00564, and the mean squared error when using the entire dataset at 0.00675, an increase of 19.68%

*2) Linear regression to predict levels of conflict as a function of global military expenditure.:* After combining the two datasets for military expenditure and conflict across the globe, we utilised Scikit-Learn's (sklearn) built in library to perform linear regression on conflict as a function of total military expenditure. The aim of performing linear regression on this dataset was to explore any potential correlations in the data, and theorising a model which could predict future conflicts based on expenditure.

In order to ensure our model was adaptable to change, we utilised cross validation and generated it using data pre-2010, and tested the mean squared error on the entire dataset, up to 2021. The weights and biases were found manually using code

This would allow us to generate a linear function which could model the number of conflicts around the world based solely on the global military expenditure of that year.

The graph shown illustrates that as military funding for a year rises, as does the rates of conflict. We can encapsulate this relationship using Pearson's r on the whole testing and training dataset, which was measured to be 0.77, establishing a strong positive correlation between military expenditure and levels of conflict around the globe.

When the linear model was applied to the data before 2010, it yielded a mean squared error of 51.93, and when using the entire dataset, the error was 58.93, showing that a simple linear model adapted fairly well to unseen data, albeit with a fairly large error. This can be attributed to many unseen latent variables that were unable to be revealed with this data. Given this large error, it may be advantageous to look into a model with polynomial basis functions, as this data is not strictly linear. When using this polynomial model on the data, with a degree of 2, we achieved an error of 36.12, however, this rose heavily to 61.73 when given the post-2010 data. This suggests that there was a lot of overfitting of the data to the training data. However, through converting our regression model to utilise polynomial basis functions, we had only increased the total testing error by 4.75%, implying that despite a large amount of overfitting on the training data, the final models, both linear and polynomial are fairly comparable with regards to their error and how they adapt to unseen test data.

We decided to push this further, and attempt a regression model with a polynomial of degree 3, however, the amount of overfitting and incapability to adapt to unseen data was so large that it was infeasible to suggest that the data follows a cubic trend.

An additional metric that we had investigated was the coefficient of determination of the models, $R^2$. In essence, it details how changes in one variable have a direct causality to the other variable, and can be used as a measure of a linear regression model's performance. The coefficient is a measure of the percentage variance in y based on the x-variable, in this case the total military expenditure. For the linear model, $R^2$ was measured at 0.578, which is a suitable score given the noise present in the data, and reflects the value of Pearson's r. With the polynomial model of degree 2, we got a similar score of 0.588. As we achieved a score greater than 0, we can state that the model does a better job than simply taking the average conflict as our predicted value, although as we do not have a score of 1, we cannot say our model is an absolutely perfect model for predicting conflicts.

### *B. K-Means Clustering to determine groups of spenders*

We used K-Mean clustering to include an alternative approach to modelling conflict prediction. Our initial thinking was to allow it to derive any possible inferences from the data. From observing the data ourselves, we inquired whether there could be a way of classifying different ranges of expenditure

into corresponding levels of conflict. The aim of this was to obtain a more general method of predicting conflict to complement our linear regression model.

Initially, we decided upon 3 cluster centers so that we could simply categorise the expenditure into low, medium and high levels. However, once we ran the algorithm it became clear that no meaningful insights could be extracted from this - the high levels of conflict could be attributed to two expenditure ranges. As well as this, the clusters varied greatly in variance and quantity of points, which suggests that they are not necessarily indicative of anything.

It made logical sense to change the number of clusters to look for to just two clusters. Once changed, it immediately became clear that there were two groups in the data that we had not seen earlier. The algorithm was repeated several times, with no change in the clusters, hence we had strong cluster groups.

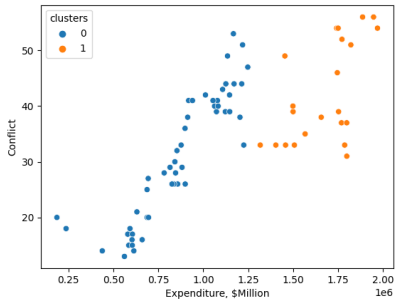



Fig. 8. Expenditure plotted against total conflict

As seen by the anisotropic cluster 0, there is a clear linear relationship between expenditure and conflicts. Cluster 1 on the other hand shows a lot of variance with no clear correlation between the two variables.

### C. ARIMA Forecasting on Time-Series Data

A limitation of using a Linear Regression model on a time series to forecast future expenditure is that the data is simply not totally linear. Although there is a general trend of the total expenditure to rise, there is a lot of noise and fluctuations within this data, which can be attributed to a multitude of factors, such as trade embargoes, economic downturn, and changes in attitudes to certain weaponry, especially towards the end of the Cold War. Due to these limitations, we decided to approach forecasting spending in a different manner, using an Auto-Regressive Integrated Moving Average model (ARIMA).

ARIMA is a forecasting model which is a combination of the differenced auto-regressive model and the moving average model [13]. The time series used for the prediction is regressed against its own past data, and the forecast error is a linear combination of past respective errors. The data is required to be replaced by differenced values in order to achieve stationary data. By manipulating the data in this way, the model uses the forecasted error in order to predict future points in the time series.

However, ARIMA relies on the data showing 'seasonality', where there are regular patterns in the data, which our data did not show. In addition, the dataset was relatively small, and the data was not stationary (the dependent variable had dependence on time, being that there is a general trend for the expenditure to increase), which meant that meaningful results were unable to be produced when using built in packages which performed ARIMA forecasting.

### D. Random Forest

To better understand the relationship between arms transfers and conflicts, we developed a model that predicts the intensity of conflict based on specific orders and the weapon class associated with it. We used the intensity of conflict from the UCDP geo-referenced events dataset as the target value.

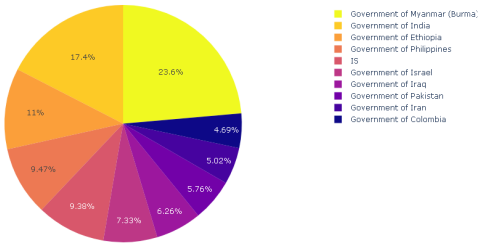



Fig. 9. Top 10 Conflict participants

To decide what to delve further into, we looked at the top 10 conflict participants and top 10 buyers (fig:2). Due to appearing high up on both of these, we selected India for some further analysis.

Fun the analysis, all the related data were transferred to one dataframe. The rows of this data contain the quantity of each of the weapon categories bought, as well as a level of conflict intensity for each year. With respect to the conflict intensity, values of '1' represent single instances of lower conflict with '2' being a large conflict.

This data was then used to train a random forest of decision trees with 5-fold cross-validation.The algorithm works by randomly selecting subsets of features and samples and training the individual decision trees on these subsets. By combining the results of multiple decision trees, random forests are able to provide more accurate predictions and reduce overfitting. After training, the average accuracy score of the model was about 0.68.

To improvement this model, we would need a more detailed dataset, specifically with regards to conflict intensity. Alternatively, combining the data used with the TIV values for arms transfer might also increase this model's accuracy.

### E. Graph Analysis

The purpose of performing graph analysis on our data was to take a mathematical approach to analysis. Initially, we

investigated with measures of centrality - where centrality represents the significance of node and is dependent on the measure used. There are three measures that were deployed: degree, closeness and betweenness centrality. To understand the results of these measures, a basic knowledge of how each of the measures interpret significance is necessary. Degree centrality assigns higher significance to nodes that have a higher number of connections to other nodes - the degree of a node. Closeness centrality observes the distance instead of degree. Instead of direct connections to other nodes, this metric uses the sum of all weights attributed to the edges adjacent to the investigated node. For betweenness centrality, this measure calculates how often a node occurs in between paths from one node to another. The algorithm to find the betweenness centrality uses shortest path algorithms to calculate the shortest path between every node and counts the frequency of nodes that lie within the path.

To deploy the measure for all nodes, a graph must be initialised using the countries as nodes and the selling/buying of weapons as weighted edges where the weight represents the value of the weapons being sold/bought. This was done using 'Networkx' [12] - a python package used to make graphs and provide graph analysis. The top ten countries for each centrality measure is shown in figure 10. Interestingly, figure 10 resembles results shown in figure 3. More specifically the degree centrality. This measure values connections to other nodes so its close resemblance to figure 3 is natural. This also suggests that these ten countries have a large connections to countries that they sell to. The closeness centrality result provides a different insight. Since this measure sums weightings from adjacent nodes, this would translate to the size of arms transfers with other countries. Betweenness centrality yielded similar results to degree centrality and has one common country to closeness centrality. This measure values the highly frequent nodes in between all possible paths in the network. This suggests that the top ten countries within this measure have high influence on the flow of transfers and could suggest that these countries are involve in intermediary transfers.

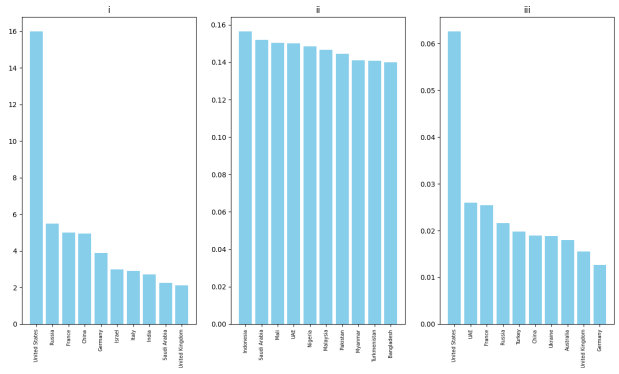



Fig. 10. Top ten countries using centrality measures where: i - degree centrality, ii - closeness centrality and iii - betweenness centrality

### F. Community Detection

After taking a look at the arms trade dataset an interesting objective arose as to check for communities throughout the countries in order to find some correlation between the buyers and sellers. Our initial approach was to compare between a Directed and an Undirected graph in order to see if the edge directions and edge weights had some influence in the community splits generated by different algorithms . However after various tries, although some of the algorithms could be used in multi directed graphs they did not take the edge direction into account.

Therefore, we decided to compare two Undirected graphs where the initial one uses all the existing edges(Graph 1) and the other one takes only the highest value edge between any two countries(Graph 2). By considering only the strongest edges, we focused on the most significant connections within the network, which helped us identify the most important communities or subgroups. In order to find the best community detection algorithm for our Graph we decided to compare various algorithms that were based in Modularity optimisation. This is given modularity measures when the division is a good one in the sense that there are many edges within communities and only a few between them [13] which makes it a key metric for community quality.

After researching community detection algorithms that optimized modularity we decided to use the greedy_modularity_communities as well as the louvain_communities which are both "Networkx" packages. greedy_modularity_communities uses the greedy optimization which starts with each node being a member of a different community and iteratively joining the pair of nodes that give the highest modularity [13]. On the other hand, the louvain_communities algorithm randomly selects a starting node and iteratively moves nodes between communities to maximize modularity.In order to decide which one to use we ran both algorithms with the most optimal hyperparameters and compared their modularity score between them:

| | Graph 1 | Graph 2 |
|---|---|---|
| Greedy_modularity | 0.333378 | 0.302537 |
| Louvain_communities | 0.333012 | 0.307703 |

Based on the highest modularity scores for each graph, We then applied these algorithms to generate communities, which can be seen in our Github. In order to visualize some aspects of the communities we decided to plot the average value of total sold weapons per country in each community as well as a table with the community sizes:

## VIII. Findings

### A. Linear Regression

Linear regression has proven to be a fundamental method in prediction. We found that as predicted, global military expenditure showed a steady and consistent rise over time. This can be attributed to a number of factors, and not necessarily just rises in conflict. The main factor that can allude to this increase is the fact that it follows a similar pattern to the change in

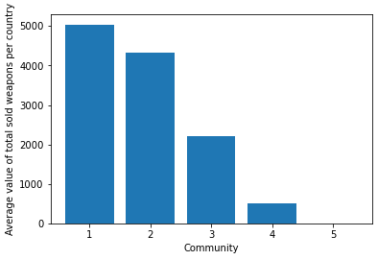



Fig. 11. Average value of total sold weapons per country in each community for Graph 1

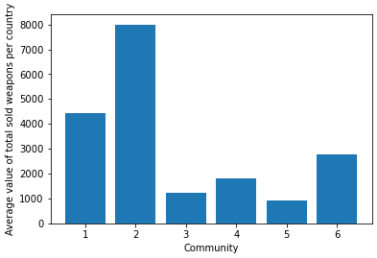



Fig. 12. Average value of total sold weapons per country in each community for Graph 2

global GDP over time. Military funding is a constant share of a country's total expenditure, so it makes intuitive sense that it will increase over time.

Indicated by the high value of Pearson's r, there is a trend in the data that if global expenditure is high, so is the total levels of conflict across the globe. Both the linear and polynomial models developed yielded similar error rates, potentially indicating that the data shows a mostly linear relationship that a polynomial model of degree 2 could also encapsulate.

Since the error rate of both models were fairly high, it is clear that these models are not optimal for accurately predicting conflict levels. If given more time to fine-tune and build upon our developed models, we could include other latent variables within our model, creating a multiple regression model. Some potential variables that we can be included are oil prices over time, inflation rates, and a measure of tensions around the globe.

As the coefficient of determination can be seen as a metric for measuring the performance of a linear regression model, we can say that the values obtained suggest that the model does a better job than simply taking the average conflict level as our estimated value. Hence using our model is an apt candidate for the prediction of conflict levels.

### B. ARIMA Forecasting

Although our attempt at ARIMA forecasting was ultimately unable to produce any useful results, it did reveal some insights about our dataset, and some further understanding that was not necessarily obvious from other machine learning models. Firstly, ARIMA models implicitly rely on the idea that the forecast data shows a resemblance to the past values in the time series. As seen in the analysis of the regression models, the data can fluctuate due to many different latent variables, such as the progression of wars. Therefore, we cannot assume that the data will show a resemblance to the past. Secondly, given that the data was non-stationary, ARIMA was not able to work efficiently given the values of the hyper-parameters for the models we had used. Ideally, we would have used further statistical tests in order to assess how suitable the model is for the data. We believe that the shortcomings of our model can be attributed to the generalisation that occurs when we considered expenditure globally. The study referenced in the related work focused on just one country, hence any noise in the data can be a direct consequence of conflicts in that country, whereas considering it through a global lens means that noise in the data is a linear combination of over 200 countries and their respective political climate.

TABLE I
COMMUNITY SIZES IN GRAPH 1 AND GRAPH 2

| | Graph 1 | Graph 2 |
|---|---|---|
| Community 1 | 71 | 80 |
| Community 2 | 71 | 29 |
| Community 3 | 41 | 28 |
| Community 4 | 11 | 18 |
| Community 5 | 2 | 18 |
| Community 6 | | 23 |

### C. Random Forest

Even though a score of nearly 0.7 was obtained from our random forest mode trained earlier, we have theorised that this model and our results are not entirely reliable.

One such reason for this is time was not taken into account when training the mode, which could lead to erroneous results as these conflicts could likely be related to the year and the events that occurred.

Another reason for this, is that the level of intensity only falls into two categories, minor (25-999 related deaths) and major (more than 1000 related deaths). These categories are rather broad and don't have a level of granularity that we want.

### D. K-Means Clustering

From our figure on the clusters present in the data, we can say that conflicts rise relative to expenditure up to a certain maximum level ($1.25 Trillion), at which point conflicts no longer rise, and after that there is no change in conflict levels. We can attribute this to rising tensions in certain areas of the world, meaning that some countries may be stockpiling weaponry.

### E. Graph Analysis

As expected, all centrality measures provided different insights into significant countries within the network. Although the results of figure 10 can be viewed separately, it may be appropriate to view these outputs all together to provide a general insight into important countries within the arms trade network.

The outputs also have cases where they closely resemble figures from data exploration which suggests that centrality can be used to provide insight on general information.

### F. Community Detection

The results for both Graphs where shown to be quite different. Firstly as shown in table I Both Graph analysis have a different number of communities as well as community sizes. One of the major differences can be shown in community 2 where the size difference is of 42 between both. This value suggests that the edge removal on Graph 2 had a significant impact on the community structure which lead to the size difference in all communities. This is further shown fig.11 and fig.12 where the average value of total sold weapons per country rose instead of dropping. We can therefore say that the even after the removal of edges the most important community was the second one as it remained intact and became even more economically active.

## IX. Website

Along with the visualisations and understanding provided by the different machine learning methods, we also decided on creating a more general visualisation in the form of website. We initially thought this website creation would be straightforward, however we encountered a few major issues along the way.

### A. Initial Creation

The initial website was built using react native. It allowed us to a decent control over the interface of our visualisation goal, since the website is dynamically updating with user inputs. The initial website included a map of the world and a simple slider that only were there for visuals and did not do anything else.

### B. Data preparation

The first step in transforming the first version of the site into the one we envisioned, was to sort through all the data we have based on a year. The two streams of data used within this section of code were,the combined dataset of all arms trade, and the Ostellus map data we gathered.

Both datasets were initially stripped of unimportant data. The remaining data was only comprised of the years the data was relevant for. This was done so that searching for this data based on the year would be more memory efficient as the data structure would be smaller.

The Ostellus data was the first to be searched. This search involved iterating through the stripped data to find the index of the whole dataset that is between the years. For each match, the name and centre of the country were grabbed from the full dataset and stored for use in the next set of checks.

The Combined arms trade data. This search was similar, in that each id was checked by date. For every item that has the correct date, the code accesses the buyer, and seller of combined set, and the borders of the countries from the previous Ostellus check.

### C. Mapping lat long to x y

First we started by writing our own function that maps latitude and longitude values into pixel values we have on our image. This method did not work robustly as it always drew on the wrong position of the canvas and did not accurately represent the ideal points.

We thought we had implemented the algorithms we researched wrong, and so we turned to directly using online implementations. There were two main implementations types that were discovered during the further research into these algorithms. Unfortunately both of these gave incorrect values on the coordinates used to test them.

Due to these complications, we turned to look for an alternative solution that would still allow us to map latitude and longitude to x and y pixel coordinates.

During researching for an alternative solution to this problem , we came across GeoTiff files. These are like normal .tiff image files except they also contain geographic metadata. The plan was to use a GeoTiff file with Geotiff.js to access this metadata. Unfortunately this library was not well documented, and after getting it to work, it turns out it was not able to achieve what we were led to believe.

After all these issues, we decided to try again creating a manual conversion function however, this time we wanted to try from first principles. Fortunately this time, the function worked and correctly mapped the latitude and longitude to the right x and y coordinates for all of the test data we threw at it.

After getting this code snippet working, we looked back at all the old functions and code we used trying to get this to work. It turned out that all of these functions had some fundamental fault.

### D. Country centre over borders

Halfway through the development process of this website, we realised it could be visually confusing to draw the borders for every country on the map. The approach would also be slower as every country would be needed to be gathered and then drawn. Our alternative was to use the countries geographical centre instead.

### E. Drawing on the web page

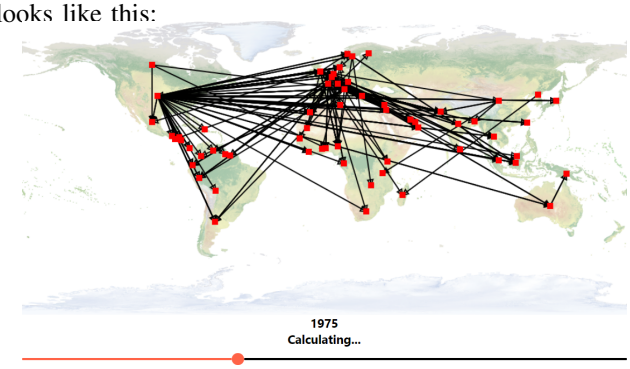
The initial solution for drawing our data on the website was to use canvases with react hooks. Hooks are simple function that allow for isolation of reusable parts. An instance of the canvas was created and put on top of the image of the map. This canvas was then triggered to update based on the year the slider was on.

In theory this all worked well, however in practice react doesn't let you call this draw function easily or in the specific area of code we needed to call it from. In order to solve this issue we had to rework how these functions were called. Moving most of the functionality of the canvas scripts inside of the main file allowed it to be called.

This relocating of code caused the canvas to no longer be rendered on top of the image like before. Due to it now being

a part of the main app's functionality, the previous fix for this issue no longer works. We had to resort to setting the background of the canvas to be our world map. This means that when you initially load the web page, no data or map is shown.

After selecting a year, you are presented with a page that looks like this:



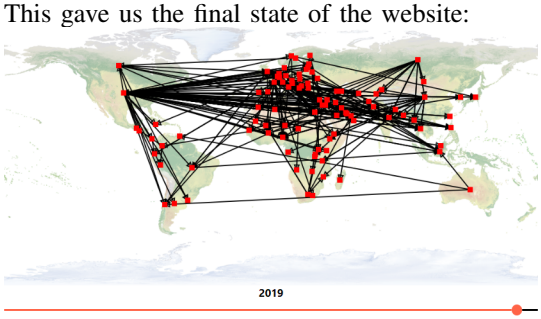

In order to explain this you need to first select a year, the website now contains a small prompt telling the user how to operate the site.

### *F. Table*

On top of displaying the map, we also wanted to show the user the exact data that was being rendered on the canvas. Presenting this in a tabulated format made the most sense.

Initial research into how to do this referenced creating custom table objects in react that can then be dynamically called. This seemed overly complicated given what we were trying to do, and so we attempted to add inner HTML to specific div id. This approach was a lot simpler, and still worked well.

This gave us the final state of the website:




| Seller | Buyer |
|---|---|
| United States | Taiwan |
| China | Thailand |
| United States | Romania |
| Israel | Germany |

We looked into hosting this site online, however we ended up running into multiple roadblocks with the JSON files not being included when running 'npm run build'. Unfortunately, there were no online solutions to this issue causing us to be unable to deploy this solution online.

Instead, users can go to the GitHub and follow the instructions on there.

## X. Discussion and conclusion

Coming into this project, we aimed to find a way to represent the causal relationship between global military expenditure, and the rates and intensities of conflicts around the world. Initial planning to create an informative visualisation of the entire arms trade network took place. This allows users to easily access information on global arms transfers.

In terms of analysis on the relationships between our data, different useful insights into what the collected data shows were provided. Despite this, there is potential for further improvements.

To further explore the forecasting of military expenditure or conflicts, it would be reasonable to further investigate different models before concluding on an accurate model. Given that there are potentially other variables to explain the levels of conflict, it would make sense to factor in these other variables, however, there may be limitations in the number of variables that would need to be included without facing the curse of dimensionality.

For graph analysis, there was an unexplored measure for centrality. This was eigenvalue centrality. This measure focuses on which nodes have a wide influence on the network. This was not deployed as it requires an undirected graph. To further include more graph analysis, algorithms that analyse the network flow could be explored. For community detection, an extension could be to generate communities every year to be visualised and implemented in the website.

For our visualisation, we were able to implement the intended plan. The map shows relevant information on trades and captures an interactive visual representation which can be traversed based on the year to be investigated. To move this forward, it may be worth adding some clarification about which square is which country. This could be done by making the squares the countries' flag .From exclusively looking at the map, it is not always possible to interpret which countries are buying and which countries are selling. Although this information is documented below the map in a table, the mentioned additional feature would be used to improved the readability of the map.

## XI. Our code

All of the mentioned code for the different machine learning methods, data gathering, and website can be found along with all the data on our GitHub Repository https://github.com/luke20332/OSINT_Visualisation

## REFERENCES


[1] K. Baker. (2023, Feb 28). "What is OSINT Open Source Intelligence?" *CrowdStrike*. [Online]. Available: https://www.crowdstrike.com/cybersecurity-101/osint-open-source-intelligence/ [Accessed: Apr. 30, 2023].

[2] European Commission. (2022, May 02). "Datastories: Open source intelligence." *European Data Portal*. [Online]. Available: https://data.europa.eu/en/publications/datastories/open-source-intelligence [Accessed: Apr. 30, 2023].

[3] D. Sharma and K. Phulli, "Forecasting and Analyzing the Military Expenditure of India Using Box-Jenkins ARIMA Model," 2020, arXiv:2011.06060 [econ.GN].

[4] Kaspersky Lab. (n.d.). Cyberthreat real-time map. [Online]. Available: https://cybermap.kaspersky.com/. [Accessed: Apr. 30, 2023].

[5] Stockholm International Peace Research Institute. (n.d.). SIPRI Arms Transfers Database. [Online]. Available: https://armstrade.sipri.org/armstrade/html/tiv/index.php. [Accessed: May 01, 2023].

[6] Stockholm International Peace Research Institute. (2012, December). SIPRI Fact Sheet. [Online]. Available: https://www.sipri.org/sites/default/files/files/FS/SIPRIFS1212.pdf. [Accessed: May 01, 2023].

[7] Stockholm International Peace Research Institute. (n.d.). SIPRI Arms Transfers Database. [Online]. Available: https://armstrade.sipri.org/armstrade/page/trade_register.php. [Accessed: May 01, 2023].

[8] Ostellus Inc. (n.d.). Ostellus Atlas. [Online]. Available: https://atlas.ostellus.com. [Accessed: May 01, 2023].

[9] Uppsala Conflict Data Program. (n.d.). Uppsala Conflict Data Program. [Online]. Available: https://ucdp.uu.se. [Accessed: May 01, 2023].

[10] Gawon Bae, Emiko Jozuka $https://edition.cnn.com/2023/04/12/asia/north-korea-missile-japan-intl-hnk/index.html$, CNN, April, 2023

[11] Helen Davidson, Aakash Hassan $https://www.theguardian.com/world/2022/dec/13/chinese-and-indian-troops-in-fresh-skirmish-at-himalayan-border$, The Guardian, December, 2022

[12] V.Kotu, B.Deshpande, Data Science: Concepts and Practice, 2nd Ed. Morgan Kaufmann, 2018, 423.

[13] Hagberg, A., Swart, P. & S Chult, D., 2008. Exploring network structure, dynamics, and function using NetworkX,

[14] A. Clauset, M. EJ. Newman, and C. Moore, "Finding community structure in very large networks," Physical review E, vol. 70, no. 6, pp. 066111, 2004, APS.

[15] Statista. (2022). Sales of the world's largest arms-producing and military services companies in 2020 (in billion U.S. dollars). [Online]. Available: https://www.statista.com/statistics/267160/sales-of-the-worlds-largest-arms-producing-and-military-services-companies/. [Accessed: May 01, 2023].

[16] Polars. (n.d.). PyPi Project Page for Polars. [Online]. Available: https://pypi.org/project/polars/. [Accessed: May 04, 2023].