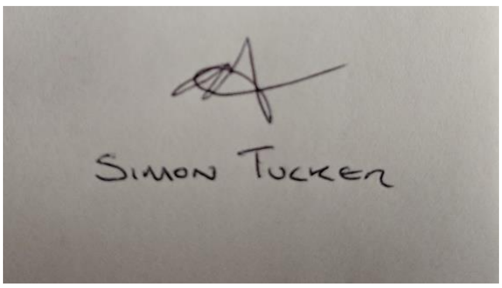
Finding Terror through Natural Language Processing and Network Analysis

Simon Tucker

Abstract:

The potential for a terror event detection system was explored which combined elements of *Natural Language Processing* and *Network analysis*. The proposed system is a feature-pivot method which detects anomalies, in networks composed of three types of detected entities (person, location and organization), as features. It would be able to detect events from a document stream or work retroactively. A pseudo-document stream was created as a corpus of documents obtained from the *Nexis News* archive. The corpus contained a variety of different publications and documents on a variety of topics. The corpus was filtered using a *Support Vector Machine* in order to remove documents which were not related to terror events. Networks were created and the output was analysed. The method was validated by using a traceable terrorism database (the *Global Terrorism Database*). The results showed that there were some issues from noise which must be overcome before the project is to progress further. There was also evidence that the dataset may have been too small, the filtering process was ineffective or that some terror events were not reported in the news. This was because there was a weak, or lack of, response for some events.

I certify that all material in this dissertation which is not my own work has been identified:



Contents

1 Introduction

In recent years there has been an increased awareness of the imbalance between the amount of data that is available and our ability to extract useful insights from that data. This is partly because much data collection is done autonomously and the sheer volume which is collected often exceeds the human, or even computational, resources that are available to process it; or we simply do not know *how* to extract useful insights. As progress is made in this area, we seem to become more aware that there is greater untapped potential than we originally considered. Many organizations, whether private or public companies [41], or government agencies [63], are looking at autonomous computational and statistical techniques to solve these problems and enhance their abilities in this regard. These techniques often come under the umbrella of Artificial Intelligence (AI). The ultimate goal is for an enhancement in how these organizations sense their environment and therefore make better decisions or actions. The term *environment* is used to refer to the scope in which the organization operates. This may include customers, suppliers, social media, the population of a country or the resources of a military battle group.

A concern that many organizations have is that when collecting such large volumes of data, information on an *event* may be available but not acted on, simply because the organization did not know about the information. The ability to *detect* these events is therefore of huge importance to the ability of organizations to make the best decisions. For example, with the growth of Artificial intelligence, we have also seen growth in the technologies used to capture data, such as optical character recognition (OCR) [59]. Some organizations may scan incoming documents into a database rather than input them manually. With the digital storage of documents comes the desire that an autonomous process could be used to inform the organization of important events that could be identified in those documents. Potentially, this could be faster, more effective and more efficient than any manual process.

Additionally, there may be new environments in which organizations may be able to sense and gain insights. We have seen increasing adoption in the use of social media [22] with projections set to rise into 2021 [5]. As the internet integrates more into the structure of our society, the separation of the 'real world' and the cyber world diminishes. This is set to persist further when we consider a large portion of the world's population have yet to gain internet access, and more household electronics are set to be connected in the future with the rise of *Internet of Things* technologies [58]. Our internet life and our real life are to become more intertwined and as this happens, the extent to which real world events are captured on the internet is increasing; digital 'footprints' which may provide potential information on past, present and even future events. This could be in the form of Tweets posted by users commenting on a celebrity's wedding or YouTube videos which were used to instigate a revolution. Furthermore, much of this data is retained on servers, readily available for people who may wish to transform it into actionable information. In short, scientists and organizations are realizing that internet environments such as social media are effective mediums in which to detect events.

Some of the earliest work in this area focussed on the detection and tracking of topics in transcribed news broadcasts [19]. Another example attempted to model the flow of information in social media and mapping how information spreads from blog posts [37]. With Twitter becoming popular in 2009 scientists were quickly interested in studying how online social networks related to geographical networks [65]. Arguably, one of the most notable works, which quite appropriately highlights the potential for event detection in social media, was by Sakaki et al., [60] from the University of Tokyo. They investigated the potential of using Tweets to detect earthquakes. Twitter data is easily accessed and in plentiful supply. Hundreds of millions of tweets are posted globally each day with adoption being relatively high in Japan. Additionally, Twitter maintains servers which update comparatively quickly. Sakaki et al., were able to implement an earthquake detection system, using Tweets, which detected earthquakes significantly faster than the current system used by the Japanese Meteorological Agency. Moreover, they could approximate the epicentre with reasonable accuracy. In another example they showed it was possible to model the path of a Typhoon with similar success.

Definition of the Problem

We have already discussed the importance of *event detection* but so far have not explicitly stated the meaning of the term. Event detection is the problem whereby some form of media (e.g. news reports, images, videos) in a collection or stream are associated with real-world events. There have been many studies on event detection using social and news media [33; 56; 61]. However, there lacks a formal definition of the problem and the meaning of the word *event* which encompasses the body of work in this area. This is because there are many applications for event detection, and many differ in their aims and context. For example, sometimes an event may be defined as the Tweets that are posted by users, or the real-world actions of which the Tweets are a result e.g. an earthquake. In previous works the term *event* is often described as something broader whereas for the purposes of this work, the focus in on a specific type of event. It is, therefore, important that a definition of *event* is drawn in the context of terrorism, where terrorism is defined as *acts committed by non-state actor*s *in an attempt to attain a political, religious, ideological or a social goal*.

In the setting of *social media event detection* there is, more or less, some consensus between several researchers that an event is something that occurs in a time and place but it is only significant if it has some kind of impact e.g. discussed in news media [17; 20; 48]. It is also true that terror events occur in a time and a place, however, they may not necessarily be reported by the news media. Therefore, we must make some alterations to this definition. Furthermore, it could be argued that defining an event as something that has an impact in the news is not particularly useful for validation or traceability. News media are not necessarily a good source of truth. There are many databases of terror events which contain more traceable data. Thus, the definition of a terror event will be defined partly by those that can be found in a reputable database, as these will be the samples on which the method will be tested. Hence, a more formal definition of the term *terror event* is provided in the *data collection* section of this thesis. Another, possible alteration to the definition would be to add that a terror event always involves some kind of entity e.g. a terrorist (organization) and a target or victim (group). Whether or not they are identified is a different matter.

Terrorism is an important subject at the moment for a number of reasons. Jenkins et al., [42] found that there has been a dramatic worldwide increasing trend in fatal terror attacks between 1970 and 2013 and, though there has been a decrease in fatal terror attacks in Europe and the United States, the number of deaths per attack has increased. Terrorists appear to be focussing more to kill in quantity. Similarly, more recent study which uses data between 1970 to 2012 from the *Global Terrorism Database* (GTD), also shows increasing trends [38]. The risk and severity of terrorism is growing and studying detection methods is important for its future prevention.

Gordon et al,. [35] suggest that people have strong concerns about the risks of future terror attacks. They asked a random sample of people to answer an online questionnaire on terrorism; most notably, they were asked if they believed it was possible that in the future a *Lone Wolf* attack (which is a terror attack perpetrated by a lone actor) could result in 100K deaths and if so to predict the year by when this event may occur. The most popular response was *yes* with an average year prediction of 2067. The reasoning behind such a large number of deaths is that as technology is advancing, or has advanced, the potential for such a Lone Wolf attacker to obtain a biological or chemical weapon, or a dirty bomb, is increasing. However, this is purely conjecture on the part of the subjects, but it highlights appropriately the levels of concern that many have about terrorism. Furthermore, Jenkins et al., [42] suggest that with the growth of the internet and social media, the political, religious, ideological or a social messages that the terrorists wish to convey through their violence, reach a broader audience. Therefore, the potential for radicalization has increased and that we are seeing more Lone Wolf type attacks. It is apparent that the challenges of counter terrorism organizations are becoming more complex and, therefore novel techniques could provide valuable information on how to counter terrorism.

News media was chosen as it is something which has not been explored as much as social media. Furthermore, news documents tend to be longer and more formal than tweets which could potentially give any methods designed to work with them a transferability advantage. What is meant by this is that the format of news reports is more similar to many formal document types e.g. letters or documents which may be scanned in using OCR. This could mean that any method that is designed to use them could likely be more easily be transferred to more mediums than one designed on tweets.

With the importance of terror event detection being discussed, the state the aims of this study are to increase the knowledge of event detection in the context of terrorism in news media.

2 Literature Review

In this section, a brief literature review of event detection will be provided as to provide a background so that the method may be explained within an appropriate context.

Much of the following has been adapted from [61] as it is the most comprehensive breakdown of event detection methods that is available.

Methods in this area can be categorized into one of three groups:

Feature-pivot	Methods of this type use a set of features within the medium. These may be specific words or imposed/hidden features or patterns. The occurrence of an event is detected by anomalies with respect to historical behaviour.
Document-pivot	These methods use clustering techniques to group documents which are similar according to a similarity metric e.g. vectorized text documents grouped by hierarchical clustering using cosine similarity
Topic Modelling	This category contains statistical methods for discovering abstract concepts e.g. 'topics', as events within the document

Event detection methods can also be appropriately categorized depending on whether they are meant to operate retrospectively or online:

Retrospective Event Detection (RED)	These methods aim to find events in an accumulated set of documents e.g. a corpus
First Story Detection (FSD)	Methods in this category are aimed to work in a more continuous online setting on a document stream

Timeslot Based (TSB) This category exists somewhere between RED and TSB and represents methods which are designed to work in a pseudo-online way e.g. incrementally.

A further attribute which may be used to group event detection methods pertains to whether they are concerned with detecting all events (*discovery*) or a specific type of event (*detection*). The below table shows the categorization of many methods in literature.

Table 1 - Categorized Methods in Literature – FEAT, DOC and TOPIC stand for feature-pivot, doc-pivot and topic modelling approaches, TXT, VIS, TM, US , SOC and LOC stand for Text, Visual, Time, User, Social links and Location modalities, and DETECT and DISCOV stand for Detection and Discovery mode. Please see Schinas et al., [61] for full references of the methods. The table has been extended with some methods discussed in this thesis and others which are further mentioned have full references in the bibliography.

Method	Pivot	Static/Stream	Modalities	Mode
Fung et al., 2005	FEAT	TSB	TXT	DISCOV
He et al., 2007	FEAT	RED	TXT	DISCOV
Mathioudakis & Koudas, 2010	FEAT	TSB	TXT	DISCOV
Sakaki et al., 2010	FEAT	TSB	TXT	DISCOV
Weng & Lee, 2011	FEAT	TSB	TXT	DISCOV
Li et al., 2012	FEAT	TSB	TXT	DISCOV
Alvanaki et al., 2012	FEAT	TSB	TXT	DISCOV
Cataldi et al., 2010	FEAT	TSB	TXT	DISCOV
Parikh & Karlapalem, 2013	FEAT	RED	TXT	DISCOV
Chen & Roy, 2009	FEAT	RED	TXT	DISCOV
Sayyadi et al., 2009	FEAT	TSB	TXT	DISCOV
Guille & Favre, 2014	FEAT	TSB	TXT	DISCOV
Zhang et al., 2015	FEAT	TSB	TXT	DISCOV
Sankaranarayanan et al., 2009	DOC	FSD	TXT, TM	DISCOV
Petrovi´c et al., 2010	DOC	FSD	TXT	DISCOV
Becker et al., 2011 [20]	DOC	FSD	TXT	DISCOV
Lee, 2012	DOC	RED	TXT, TM	DISCOV
Petrovi´c et al., 2012	DOC	FSD	TXT	DISCOV
Moran et al., 2016	DOC	FSD	TXT	DISCOV
Melvin et al,. [49]	FEAT	TSB	TXT	DISCOV
Cui et al,. [29]	Doc	TSB	TXT	DETECT
Moutidis & Williams [52]	FEAT	TSB/RED	TXT	DISCOV
Aggarwal & Subbian, 2012 [17]	DOC	FSD	TXT, SOC	DISCOV
Becker et al., 2009	DOC	RED	TXT, TM, LOC	DISCOV
Becker et al., 2010	DOC	RED, FSD	TXT, TM, US, LOC	DISCOV
Reuter & Cimiano, 2012	DOC	FSD	TXT, TM, LOC	DISCOV
Petkos et al., 2012	DOC	RED	TXT, VIS, TM, US, LOC	DISCOV
Bao et al., 2013	DOC	RED	TXT, VIS, TM, LOC	DISCOV
Wang et al., 2012	DOC	RED	TXT, TM, LOC	DISCOV
Petkos et al., 2017	DOC	RED	TXT, VIS, TM, US, LOC	DISCOV
Benson et al., 2011	TOPIC	RED	TXT	DETECT
Ritter et al., 2012	TOPIC	RED	TXT	DISCOV
You et al., 2013	TOPIC	RED	TXT, TM, LOC	DISCOV
Zhou & Chen, 2014	TOPIC	RED	TXT, TM, LOC	DISCOV
Zhou et al., 2015	TOPIC	RED	TXT, TM, LOC	DISCOV
Cai et al., 2015	TOPIC	RED	TXT, VIS, TM, LOC	DISCOV
Diao & Jiang, 2013	TOPIC	RED	TXT, US	DISCOV
Wei et al., 2015	TOPIC	RED	TXT, TM, LOC	DISCOV
Hu et al.,	TOPIC	RED	TXT	DISCOV
Bao et al., 2013	DOC	RED	TXT, VIS, TM, LOC	DISCOV
Wang et al., 2012	DOC	RED	TXT, TM, LOC	DISCOV

The end goal of the terror event detection method is to be able to detect an event close to when it is reported by a document. One challenge that is associated with this is that there will be noise in the data from previous events. Documents about the same event do not all occur instantaneously but instead a particular event may be discussed in the media for a great length of time through multiple documents. This means when new documents are published about a novel event, there will also be documents being published on previous events which creates noise. A way to reduce this noise is to consider, not just the documents which are published at the present time, but some which were previously published in a time window which extends some distance into the past. This allows us to take into account any trends or historical behaviour. A perhaps, simple way to approach this would be to measure the average raw count of a particular word in the time window and if the current count of the word exceeds a threshold of a certain number of standard deviations from the average, consider it an event. Melvin et al,. [49] proposed a TSB feature pivot method which mines phrases from documents within a time window and then builds a phrase network. Instead of monitoring word frequencies, attributes of the network are monitored and this allows them to better recognize the more abstract anomalies in the documents. Using networks allows the method to take into account relationships between words or phrases.

Moutidis and Williams used a similar approach but extended the method to include entities such as *persons*, *locations*, and *organizations* [52] which are extracted by Natural Language Processing (NLP) techniques. This is quite a logical progression as we have already discussed in our definition of a terror event is that it will also involve entities of the same type. Their method is also designed to work in discovery mode but could possibly be adapted to detection mode.

An initial choice that has to be made concerns a fundamental aspect of building a terror event detector. This is the decision at which point in the process does it become specific towards terror events i.e. *detection mode*) rather than *discovery mode*. There are two candidate options for how this may be implemented: process a corpus of documents and look for features in the network structures or clusters that are produced or look for features in the incoming documents before processing occurs and filter terror related documents. The latter has the advantage that it is going to be less computationally demanding as it must process less documents. An important consideration is which method to use to filter terror event related documents.

A *detection* mode method which attempts to detect events of disease outbreak by using social media was proposed by Gomide et al, [34]. Their method detects outbreaks of Dengue, which is a mosquito-borne viral disease. It requires the construction of a dataset by which tweets are manually split into sentiment categories: personal experience, ironic, opinion, resource (informative) or marketing. This dataset is used to train an associative classifier which maps a new tweet to one of the six categories; so the method is essentially a document pivot approach. The process of detection initially samples tweets which contain the word *Dengue*, the classifier step acts as a filter of sorts which helps to assess the severity of the detected event. It is worth noting that a particular tweet may be assigned to multiple categories simultaneously. Therefore, instead of predicting a tweet's association with a single class, they use a scoring system which estimates the likelihood of a certain sentiment being the implicit attitude of the mentioned tweet. The next step uses a linear regression model that had been fit using the volume of tweets containing the word *Dengue* and high levels of personal experience sentiment, with the intention of predicting the actual number of cases which were reported in a ground truth dataset obtained from the Health Ministry (Brazil). The idea is that the volume of tweets with high levels of personal experience sentiment will more closely match the volumes of Dengue cases. One issue with applying this method to terror event detection is that it may be difficult to categorize documents in a similar way as they have done with Dengue.

Another issue relates in the use of the search term *Dengue*. For terror event detection, search terms would have to be chosen carefully as inappropriate search terms will miss important documents and so reduce the sensitivity of the detector. Conversely, a larger number of search terms will increase the number of incoming documents and consequently, the computational demand and noise. Searching for the names of known terrorists and/or perpetrator groups is an option but new terror events can be implemented by, as of yet, unknown actors. Furthermore, perpetrator groups are sometimes identified after the event has occurred or not at all. Search terms which are likely to appear in documents reporting on terrorist events are therefore necessary. Some candidate examples of these include: *terror attack* or *terrorist*. However, there may be differences in the vocabulary in how terror events are reported and there may not exist a simple set of words for this task. Another consideration is that there may be little benefit in developing a system that relies on a search engine of a database. There is more potential for benefit if a system can extract relevant articles from any news database.

Cui et al,. [29] proposed a document pivot FSD method for detection of foodborne disease events using social media. Their method searches for specific tweets that contain keywords related to foodborne disease and then samples the tweets from a time window which extends both sides of the identified tweet of interest. Although, it is not enough to select all tweets in the time window, but better to use a similarity measure on the vectorized tweets and take only those tweets which have a similarity measure which exceeds an imposed threshold. In their implementation, they used an open source software toolkit named *Word2Vec [49]*, which is a type of neural network trained on roughly 100 billion words from a Google News Dataset. It can project words into a vector space which allows them to use cosine similarity as their similarity metric. They claim that sampling in this manner captures more useful and related tweets rather than simply sampling tweets based on keyword matching. Vectorizing documents could be a good approach to take with terror events because it does not require as much supervision. It is only required that a set of positive documents and negative documents are created. Then a binary classifier could be used to separate out the unwanted documents. A *discovery mode* method could then be used to the simply detect events in the remaining documents.

3 Data Collection

Two datasets were required for this task. A set of ground truth data for comparing the results and validate the performance of the detector. This would come from a highly regarded and reliable source that collects details of incidents of terrorism. A second set would comprise of a corpus of news articles. This would be processed by NLP and then network analysis. It would be split into a validation set and a test set. The former would be used for tuning any hyper parameters and maximizing the performance of the detector. The latter was used to test the final performance.

Ground Truth

The ground truth data was downloaded from the Global Terrorism Database (GTD) provided by the National Consortium for the Study of Terrorism and Responses to Terrorism (START) [7]; which is a University of Maryland-based research and education centre comprised of an international network of scholars committed to the scientific study of the causes and human consequences of terrorism in the United States and around the world. The GTD contains, as claimed by START, over 180,000 recorded, transnational and domestic, events of terrorism. These are easily searchable by a number of criteria such as number of fatalities or perpetrator group. The data was originally collected by Pinkerton Global Intelligence Services for clients that were interested in surveying the risk of terrorism in different countries. A good exploration of the database comparing it to others, including its faults can be found here [32].

Another database that was considered was the RAND database of Worldwide Terrorism Incidents (RDWTI). This is another database of terrorism incidents but is downloadable as a single spreadsheet. Though, the database is much smaller and the most recent events occurred in 2009. However, the GTD is a much more comprehensive database and contains more recent events. Another, possibility was the *International Terrorism: Attributes of Terrorist Events* dataset (ITERATE) [50], which is quite similar to the GTD. However, it only includes transnational terror events and not domestic. As it was not necessary to apply this constraint on the data at this stage, and because it will be difficult to only find news articles about transnational terrorism, it decided that it was not the most appropriate.

The GTD data for the years 2018 and 2019 were incomplete and it was difficult to judge where events may be missing or may be uploaded in the future. The most recent full years' worth of data, which was available, was for the year 2017. Considering this, and that using the most recent data available is going to better ensure that the experiments are going to be more representative of a present-day application of the system being designed, it was decided 2017 was a good compromise.

Definition of Terror Event

The aim of the experiment is to aid towards the design of a terror event detector and, although it may seem obvious to some what is meant by the term *terror event*, it is important that a definition is provided as to reduce ambiguity. An explicit definition would allow us to make easier choices throughout the experiments, be clearer in the reporting of the results and increase reproducibility. Furthermore, terror events are somewhat frequent. In fact, for 2017 the GTD has recorded 10900 separate incidents. A more explicit definition may reduce the number of incidents to a more manageable size. Focusing on fewer incidents allows errors to be more noticeable and this can benefit the quality of what is learned at the end. One easy way of doing this is to only include events which have occurred in a specific area. For example, one may wish to only consider events in Europe or a specific country. However, this may not be practical because it is more difficult to restrict news articles to a specific area. Therefore, doing this may add unnecessary noise from events outside of the chosen area.

START are, arguably, somewhat liberal with what events can be included in the GTD, which is evidenced by there being such a high number. They give little indication of what criteria an event must meet to be included other than the statement '*it does not include foiled or failed plots, the distinction being that the attack must actually be attempted to qualify for inclusion in the database. Likewise, the GTD does include attacks in which violence is threatened as a means of coercion but does not include threats to attack where no action is taken*'. As well as the typical search options available with most search engines, the GTD provides three criteria, which a user may enable in order to narrow down a search. When enabled, terror events must then meet these criteria to be included in the search results. It was decided to include these criteria in the definition used in this thesis because it allows us to be more explicit about what is contained in the data. They are included as the first three criteria in the definition used in this project. The full definition for what is considered a *terror event*, for the purposes of the experiments, are henceforth:

Criterion I: The act must be aimed at attaining a political, economic, religious, or social goal.

Criterion II: There must be evidence of an intention to coerce, intimidate, or convey some other message to a larger audience (or audiences) than the immediate victims.

Criterion III: The action must be outside the context of legitimate warfare activities, i.e. the act must be outside the parameters permitted by international humanitarian law (particularly the admonition against deliberately targeting civilians or non-combatants).

Criterion IV: The action is executed by a non-state entity i.e. not a nation.

Criterion V: The action must have been attempted and is therefore not a foiled attempt.

Criterion VI: The action occurred at a specific time and location.

The below figure shows a breakdown of events per region, according to the GTD, over the year 2017. The way the regions have been defined is as they are grouped in the GTD. It can be seen that the regions with the highest number of incidents, by a considerable margin, are the *Middle East & North Africa*, and *South Asia*. After that, sub-Saharan Africa, then Southeast Asia. It can be seen that Eastern and Western Europe, and North and South America experience far less terror incidents. Australasia & Oceania, Central America & Caribbean, Central and East Asia are approximately 100 or lower.

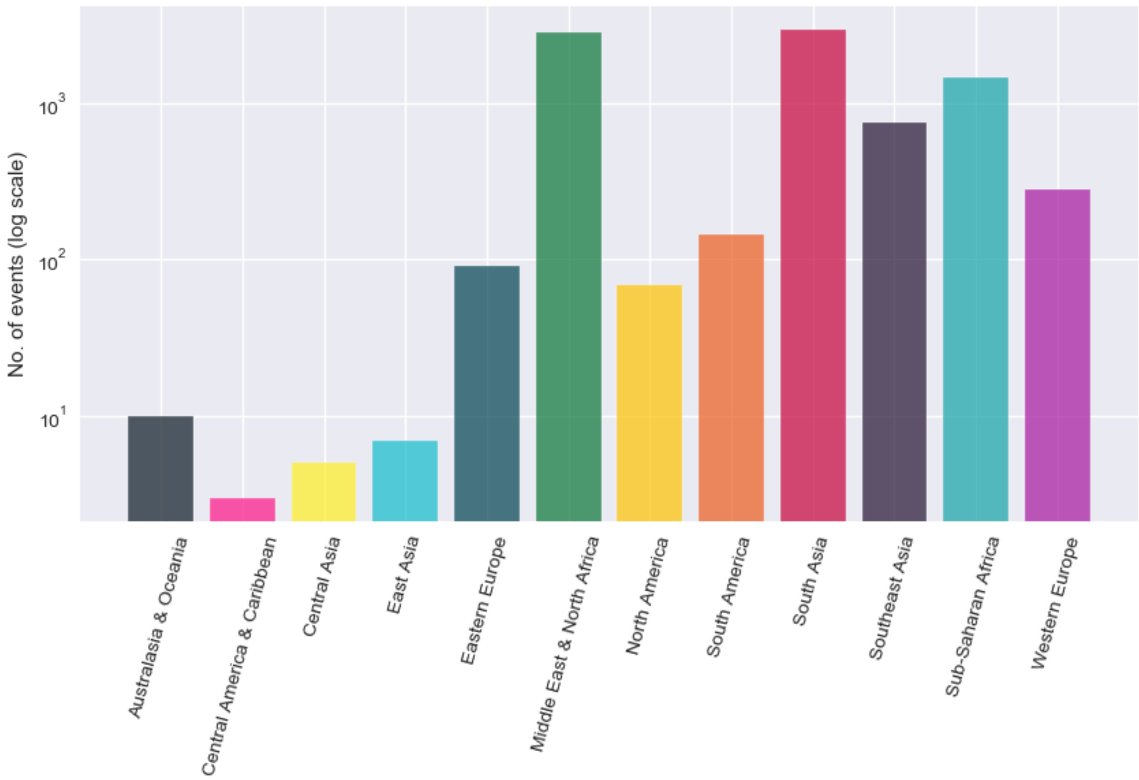


Figure 1 - Terror Events by Region 2017

News Corpus

To simulate a document stream, a corpus of documents was obtained from the Nexis news archive which is provided by LexisNexis [8]. Nexis is an extensive news database containing many text news formats such as websites, newspapers and newswires.

Whilst performing a survey of the Nexis database it was observed that there were significant differences between what is considered terrorism in the GTD and what is considered terrorism by the news media. It was evident when attempting to find reports of specific events, that the terminology and volume of reports could differ considerably depending on the location of the event. One of the most noticeable patterns of this inconsistency was in the reporting of troubles occurring in the Ukraine where perpetrator groups are often pro-Russian militants. These events are recorded in the GTD as terror incidents, but the news media tends not to refer to them as terrorism. At the least, this was evident in the English language articles that were surveyed, and admittedly the news media in Ukrainian or Russian may have been more explicit in its descriptions. Reports of these events were often noticeably more difficult to find which could be due to a few reasons. The vocabulary is different, for example the words 'terrorist' or 'terrorism' were scarcely used. There may be political reasons why English language news media does not consider these events to be terrorism, such as the countries where most English language publications are located have tensions with Russia, which is a topic which will not be explored in this work. Another reason could be that terror events occurring in these locations are not as shocking, and therefore 'newsworthy', to English language publications or that the geographical or communication/ language distance affects the degree to which these publications are sensitive to these events. And these reasons could lead to less reporting on events in these regions.

Differences in the coverage of terror incidents has been previously studied and researchers have observed many similar patterns. A study by Kearns et al., [45] highlights that terror events are underreported in general but an event is more likely to be reported if the perpetrators are Muslim. Furthermore, with differences in severity of an incident there are also differences in reporting. An event which has many fatalities often impacts the media more profoundly than an incident which has no fatalities.

Another factor which may affect the coverage of a terror attack is the potential that the coverage of an event may provoke, incite or inspire more attacks. There is strong evidence that media attention can increase the severity and quantity of terror attacks [21]. The aim of terrorism is to express a political ideology through acts of violence and media coverage provides a platform for terrorists to reach an audience. The media and terror organizations are, to an extent, both actors in a somewhat volatile cycle [43]. Furthermore, there is the potential that coverage of an event may encourage copycat aggression or incite a retaliatory response. Reporters have advice available on how to report certain subjects so that their reporting does not produce unwanted consequences [12]. It is possible that some events may be less apparent in the media for these reasons.

It is apparent that news media is a somewhat bias medium in which to detect terror events. A fundamental challenge of detecting terror events through news media is, therefore, to overcome the challenge of unequal reporting. Some events will likely be more difficult, or even impossible to detect if they are not reported. This is potentially even more difficult when one considers that terror events which have a large impact in the news, may create noise which could mask lesser reported events. The initial survey found references and discussions of the 11 September 2001 attack on the World Trade Centre in some of the material from the 2017 corpus which shows how an event which has high impact could potentially add noise to the problem.

Considering that terror events are reported in such an imbalanced way, a selection of publications were chosen based on the following criteria. A balance of political orientations as to attempt to reduce political bias. It was also desirable to get a range of publications from different countries for the same reason, reduce other variances and increase the probability that one may capture more lesser reported events in areas which may not be reported in popular UK publications. For these reasons international publications *Associated Press International* (API) and *United Press International* (UPI) were chosen. It was also observed that *BBC Monitoring* and *Ukrinform* often reported on events which were missed by UK press, so it was decided to include these too. Nexis news archive puts some constraints on how certain publications can be downloaded. This means that some publications require search terms and it is not possible to download an entire year of their published material, in a given time window. As it was desired to have a corpus which simulates a news stream, it was decided not to include these publications. The corpus, therefore, contains all the published material of the chosen publications in the year 2017. The chosen publications and their contributions to the corpus can be seen in the below figure. In total there were 0.52M articles from websites or newspapers, with an average word count of 491 words. Additionally to those previously identified, the other publications were *Agence France Press*, (AFP), *CNN.com*, *EuroNews*, Sky News, *The Guardian*, *telegraph.co.uk*, *thesun.co.uk* and *thetimes.co.uk*.

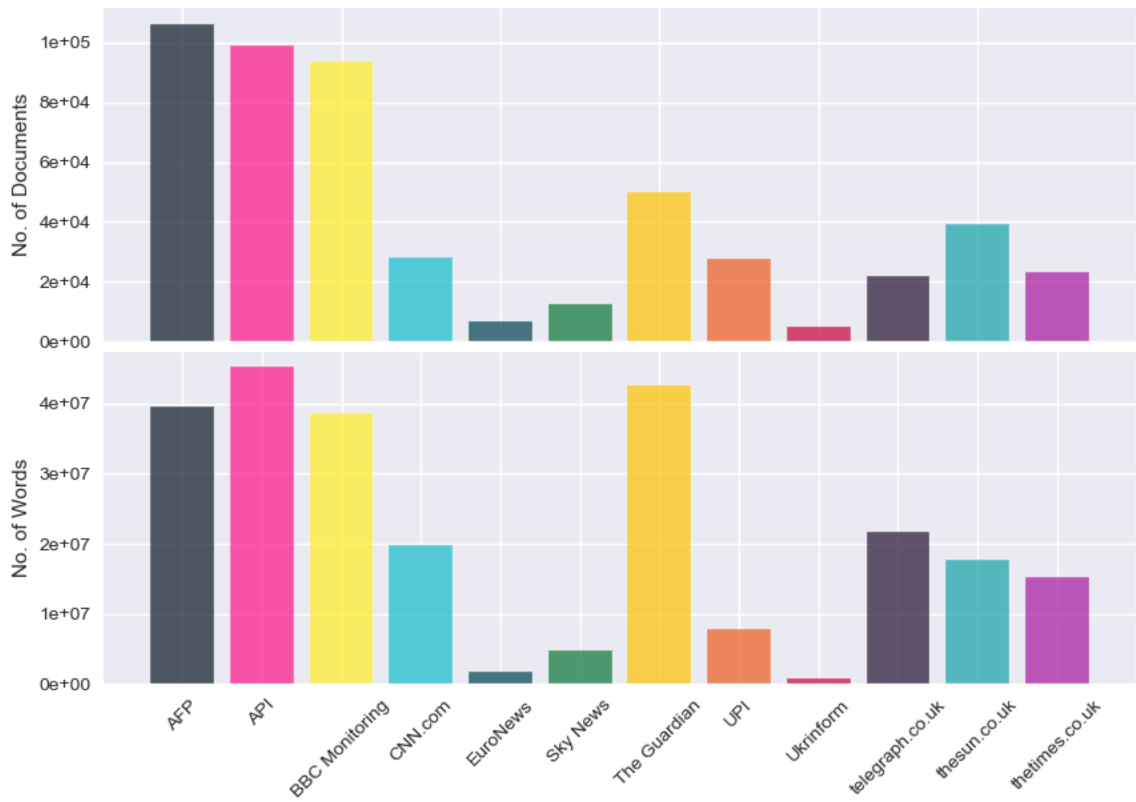


Figure 2 - Contribution of Each Publication to the Corpus

4 Method Development and Experimental

The method development and experimental sections have been combined because the entire method has required some experimentation in order to select the best approaches and it makes logical sense to present the development processes alongside the experimentation used to justify each step.

The main outline of the process that has been implemented can be seen in figure 3. The intention is that the process is a feature-pivot, *detection mode*, TSB method and will be used on a document stream, so it is *intended* to eventually be a TSB method although technically all of the testing and training is done on past data in a corpus, so is retroactive. The method will detect features in network structure built on three types of entity *persons*, *locations*, and *organizations*. At some intervals the same process may have also been applied to the ground truth data; this will be stated if so. The main outline of the process is to vectorize all documents which come into the pipeline from a document stream using a Doc2Vec model. Once vectorized, each document is classified as relevant to a terror event or not relevant by Support Vector Machine (SVM). Relevant, un-vectorized, documents then proceed to the pre-processing stage where they are cleaned. The next stage is Named Entity Recognition. Once the documents have been searched for entities, a disambiguation step is performed. The final stage is to build Networks and perform analysis. The intention is that in future work the process will be extended into a full event detection method.

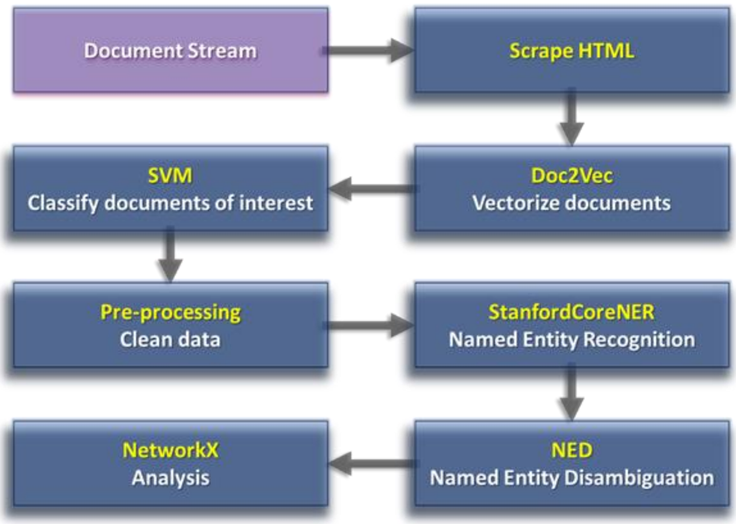


Figure 3 - Method Flow Diagram

Hardware and Software

All processes and software development were performed using an Intel i7 5820K (six core, 12-threaded, 3.3 GHz) CPU with 16 GB (2133 MHz) RAM. This was adequate for this project as most processes were only single threaded and memory usage was well within that available.

The main programming language that was used, unless otherwise stated, is Python 3.7.1. Numerous libraries and APIs have also been used which will be stated in each relevant section.

Scraping HTML

The Nexis news database allows one to download chosen documents in several different formats. However, once downloaded they would have to be scraped into a useful data structure. Experiments were performed with textfile (.txt) and HTML and it was decided to use HTML because the documents tended to be more consistent in their formatting, which would increase the quality of the scrapes and consequently, any processes that occur after the scraping procedure. The main difficulties were matching the correct parts of an HTML file. For example, identifying when one document ends and another begins, the title and date etc… Furthermore, it was noticed that formatting differs between publications and adding new publications to the corpus could require some alterations to the scraper that would be used. It was decided that the Selenium Python library [14] was suitable, as it is familiar and there are few other criteria the scraper must meet. Selenium is an experimental webdriver designed to autonomously control a browser. However, it was used in a 'headless' mode whereby no browser is required. In hindsight, although the choices of HTML and Selenium worked well, the process was quite slow. Scraping textfile documents was much faster and there are also other HTML scrapers options which may be much faster.

The textfile and HTML downloads contain a line at the start of each document that reads '*DOCUMENT X of Y*' where X is the number of the next document out of Y documents. Python's built-in regular expressions module was used to match this line and identify when a new document was beginning. Furthermore, the Y value is used to make a loss check, at the end of the scrape of each file, to ensure the number of scraped documents matches Y. An issue that had to be overcome was that a small number of documents may be missing from the file and there would be instead the error message '*We are sorry but* *there is an error in this document and it is not possible to display it*'. This was identified as the loss check would raise an exception. Duplicate downloads of a file with the error would also contain the error so it is hypothesized that the error is something related to the Nexis database or how they generate downloads. By this time, corpus had already been downloaded and the scraper had been coded to work with HTML so it was not assessed whether the error was exclusive to HTML downloads. The scraper was updated to account for the error message by updating $Y = Y - 1$ each time it was encountered.

A *regular expression* was also used to match the date and time near the start of a document. Once a date and time was located, the fuzzy datetime parser in the Python *dateutil* library [3] was used to extract the date and time into a *datetime* object. The *pytz* [11] Python library which has a wide-ranging database of timezone codes was also used. Otherwise, when a date and time string in the document contained a timezone code, the parser could throw an error or extract an inaccurate datetime. Furthermore, this allows us to resolve all of the timezones to *Coordinated Universal Time* (UTC), which is the international timezone, and therefore allow us to make more accurate comparisons of documents' publishing date and time which allows more accurate sorting. A notable observation was that web articles always contained a time, precise to the nearest minute, whereas newspaper articles often did not. The web articles therefore produced a more precise datetime object and this may be useful for any future analysis involving time. For newspaper articles, the time would be extracted as 00:00, as no time information was available. Once all documents were extracted, single list containing dictionaries was obtained, where each dictionary was a document containing all of the information of the document, such as publication, datetime, author, headline and the content. The reason for this data structure is that it allows us the sort and slice easily, which will be useful for operations that require a sliding window.

Vectorizing Documents by Doc2Vec

Doc2Vec [47] is an extension of the better-known Word2Vec [51] models. Word2Vec refers to a group of two-layered neural networks trained to produce word embeddings. During training, they take as input, a corpus of text and produce a vector space whereby each unique word is assigned a corresponding vector in the space. Where Word2Vec allows the projection of a single word into a vector space, Doc2Vec extends this ability to whole documents. Basically, Doc2Vec allows, to an extent, to assign numerical values to a document's meaning. Or more explicitly, coordinates in a vector space. The documents can consist of singles words, sentences, paragraphs or multi-paragraph texts. The attraction of Word2Vec and Doc2Vec over other methods of vectorization, is that they both model some semantic information. This means that one can expect texts with semantic similarity to exist in closer proximity in the vector space than documents with less semantic similarity. In fact, Word2Vec and Doc2Vec are models of the relative semantics of words and text. What is meant by this is that the actual vector a word is assigned is arbitrary but its position in relation to other words is important. To train them, they take as input a large corpus of text and learn the relative meanings of the words, not the absolute meanings. Once a document is projected into a vector space, a classifier may be able to distinguish between terror event related documents and everything else.

Two pre-trained Doc2Vec models obtained from [4] to vectorize documents. The difference between the two models is that the first model (WIKI) was trained on the full collection of English Wikipedia, which surmounted to approximately 35M documents and 2B tokens at the time; the second model (AP) was trained on Associated Press articles from 2009 to 2015 and numbers approximately 25M documents and 0.9B tokens. The justification for using pre-trained models is that it is unlikely that it was possible to get training sets as large and train the models as well within the resources of this project. Furthermore, the models have been reasonably well validated in literature; full details on the training and evaluation processes can be found here [40]. Another consideration is that the corpus contains a large amount of Associated Press articles which may mean the AP model is especially able to transfer its learning to the task. Both of the models project into a 300 dimensional vector space.

For the pre-trained Doc2Vec models to work, it was necessary to use Python 2.7 and a forked version of the Gensim Python library created by the same authors [4]. The models also required a compatible C++ compiler, which on windows, was Visual Studio C++ 9.0 for Python 2.7.

For pre-processing, Gensim contains its own tokenizer which essentially strips all punctuation from a string and splits it into lowercase words. A few trials were run with removing stopwords before using Doc2Vec and no measurable difference in the performance of the classifiers that were trained from the output vectors afterward was noticed. Furthermore, removing stopwords is not mentioned in the documented usage of Doc2Vec [40; 47]. Therefore, it was decided not to remove stopwords.

Filtering by Support Vector Machine

To train a classifier to act as a filter, a set of labelled documents is required. Further documents from Nexis which were by the publications in the main corpus were downloaded. This was to reduce the effect of any vocabulary and formatting differences between publications which may affect the performance of the classifier. Furthermore, the documents were all pre-2017, so that it can be ensured that the training data contains no documents from the corpus. Otherwise, this may affect the performance of the main process and assessments of its performance will be invalid.

Caution had to be taken with which documents were selected as to represent the populations of both classes in the 'real world'. For the negative class, this meant downloading documents which were about many different subjects that were not terror events. Another aim was to get a large portion of documents that used similar language to terror event documents. For example, news reports of war or conflicts that were not terror related, murders, reviews of movies where special effects were discussed or even sports. Ideally, documents where the words terrorism or terror attack is mentioned but the document does not contain information about a specific terror event (e.g. perhaps in discussing a TV series or political manifesto). The reason for doing this is so that the classifier is not learning to simply distinguish between the occurrence of certain words, such as *terrorism*, *attack*, *explosion* or *killed*. It is desirable for the classifier to distinguish between more abstract concepts e.g. the reporting of a terror event and anything else. In fact, one of the motivations for doing this was that, after observing how the SVM classifier had classified the training set, it appeared it was separating documents exclusively on violence). For the positive class, one aim was to download documents on terror events that were perpetrated by many different organizations, in a range of countries and which used a range of vocabulary. Another consideration was to use only one document per event. The idea of this is that it reduces the similarity between training and validation data or training and test data, as neither pair can contain documents which are about the same event. This means that the performance metrics better reflect the ability of the classifier to generalize to unseen data. In total 4863 documents were downloaded, 3656 negative and 1207 positive.

The performance of three classifiers using both Doc2Vec models, so six classifiers in total, were trialled. The three chosen classifiers were Random Forest [27], AdaBoost [31] (ensemble methods which used decision tree classifiers as weak learners) and SVM [25]. The implementations of the classifiers were from those available through *Scikit-Learn* [13]. The classification training data was shuffled and 20 % was randomly selected and put aside for final testing. Parameters were tuned using 5-fold cross validation.

SVM is an algorithm by which a single optimal boundary which separates the two classes is found, a separating *hyperplan*e, by maximizing the width of the boundary's parallel margins, between the hyperplane and samples, on either side. It is a convex optimization problem. The algorithm can model linearly inseparable data by projecting it into a higher dimensional vector space by using a non-linear kernel such as the Radial Basis Function (RBF) where the data can be separated linearly. A hyperparameter *C* controls the trade off between margin width maximization and violation of the margin by samples. An SVM with low C is a soft margin classifier and allows more violation of the margin and therefore a wider margin, conversely high values of C lead to thinner margins and less violation. The SVM used was the *SVC* class in Scikit-learn, set to probability mode so that the output is an array of probabilities calculated by *Platt Scaling* [57]. This yields more information on the distances of data points from the decision boundary. It also allows us to use performance metrics which require probability rather than predictions.

Principal component analysis (PCA) [44] was used in order to see the distribution of variance among the classes. PCA is a technique used for dimensionality reduction. It can be seen in the figure 4 that the contribution of each dimension to the variance does not diminish greatly as dimensionality increases. Dimensions with higher variance have a higher probability to contain information that distinguishes between the two classes. It may have been possible to drop approximately 25 dimensions from the data and still retain 95 % of the variance. However, the dimensionality reduction can result in loss of information and it was not a valuable compromise for this project; the dataset was small enough to fit in memory for training processes, therefore, it was not used. However, it should be a consideration, that if training a new Doc2Vec model, to use more dimensions as the curve in the below plot appears as though it would extend further upwards with more dimensions.

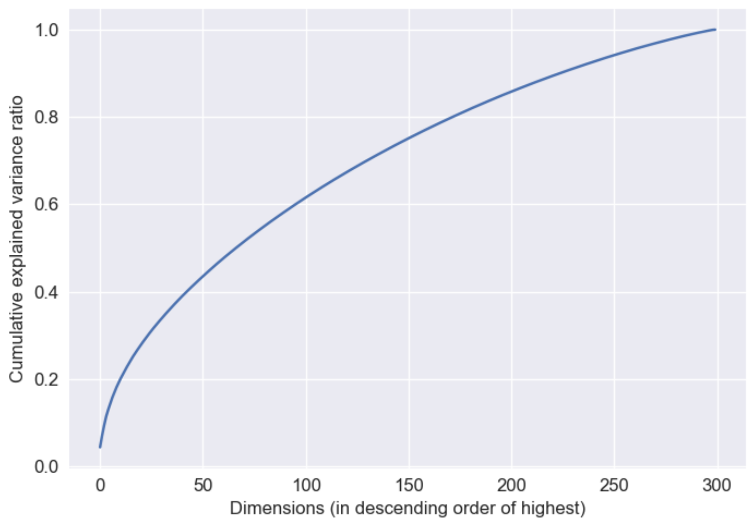


Figure 4 - Explained variance of dimensions in Classifier training data by PCA

In figure 5, the *scitkit-learn* implementation of Stochastic Neighbour Embedding technique (t-SNE) [64] has been used in order to visualize the data. This technique reduces the dimensionality of the data to few enough dimensions that it may be visualized. It is difficult to interpret the visualization as the method makes drastic alterations to the data for this process to be possible. Furthermore, there are many hyperparameters to tune, there is an element of stochasticism and the technique is sensitive to noise, which can produce very different structures in the data each time it is applied. However, what can be inferred from the visualizations is that, after experimenting with many different combinations of parameters, the different classes are always in different regions which suggests that there is some intrinsic difference between the two classes. It should be noted that the dimensionality was reduced to 50 by PCA beforehand as to reduce noise.

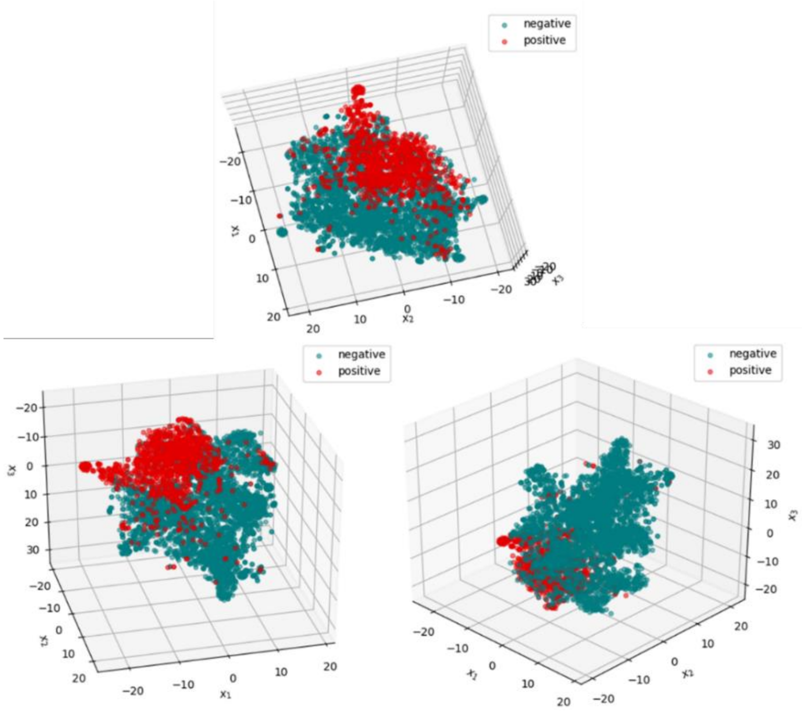


Figure 5 - Vectorized Documents by t-SNE

The metric which was maximized during cross validation was the Area Under the Curve (AUC) of the Receiver Operating Characteristic (ROC) curve [26]. The ROC is a graph of the performance *true positive rate* (TPR) versus *false positive rate* (FPR) at all classification thresholds (1, 2). A perfect classifier will yield an AUC score of 1 whereas random choice will yield 0.5. A score of 0 would suggest that the labels are reversed, or at the least, they *could* be reversed to make it into a perfect classifier. In the case of SVM, one can imagine that it is equivalent to, for a particular model, moving the decision boundary along the line which is its perpendicular bisector and measuring the TPR and FPR when the line is on a sample. More specifically, a two-dimensional array is constructed where each row records information about a single probability estimate of the positive class from the ouput of a classifier. With the information being, which other probability outputs were higher (or equal) and which were lower, denoted 1 or -1 respectively. Then for each row, TPR and FPR values are calculated, as in equation, by comparing each row with the true labels. The result of this is then sorted in ascending order of FPR. These are plotted with TPR on the y-axis and FPR on the x-axis. This yields the ROC curve, of which its integral is equal to the AUC (3). There are many ways to interpret AUC such as: the expectation that a uniformly drawn random positive is ranked before a uniformly drawn random negative or the expected proportion of positives ranked before a uniformly drawn random negative.

$$TPR = \frac{true\ positives}{true\ positives + false\ negatives} \tag{1}$$

$$FPR = \frac{false\ positives}{false\ positives + true\ negatives} \tag{2}$$

$$AUC = \int_0^1 TPR(FPR)\, dFPR \tag{3}$$

Table 2 - Performance of Classifiers

Classifier	Doc2Vec Model	Class	Acc.	Precision	Recall	f1	AUC
SVM (test)	WIKI	-	0.95	0.96	0.97	0.97	0.98
		+		0.91	0.88	0.89	
SVM (cv)	AP	-	0.94	0.95	0.97	0.96	0.97
		+		0.91	0.85	0.88	
	WIKI	-	0.94	0.96	0.97	0.96	0.99
		+		0.90	0.87	0.89	
Random Forest (cv)	AP	-	0.85	0.84	0.99	0.91	0.95
		+		0.96	0.41	0.57	
	WIKI	-	0.87	0.86	0.99	0.92	0.96
		+		0.95	0.51	0.66	
Ada Boost (cv)	AP	-	0.88	0.86	0.99	0.92	0.97
		+		0.97	0.52	0.68	
	WIKI	-	0.90	0.89	0.99	0.94	0.97
		+		0.95	0.62	0.75	

**test* and *cv* denote performance during cross validation and testing respectively

It can be seen in the above table that the WIKI Doc2Vec model produced higher performance over the AP model. This could be attributed to the WIKI model being trained on a considerably larger corpus of documents and, therefore, could have seen a much larger vocabulary of words. Another possibility is that the corpus consists of publications other than API and the AP model may not have encountered a significant portion of the vocabulary in the corpus. The Doc2Vec models ignore words which they have not encountered before rather than attempt to extrapolate a meaningful vector.

The best performing classifier during cross validation was the SVM (table 2) with hyperparameters *C* = 4.096 and *gamma* = 0.0512 and *RBF kernel*. The SVM also achieved equivalent performance on the test data which was a good indication of its ability to generalize. It was apparent that the Random Forest and Ada Boost classifiers had a higher tendency to misclassify the positive class (terror event related). The confusion matrix of the SVM on the test data can be seen in figure 6 and shows, although the misclassified samples of each class are roughly equal, there is a slight bias to classify unknowns as negative.

After using the SVM to filter the corpus there were 59K positively classified documents. The distribution over 2017 can be seen later in the thesis (figure 7).

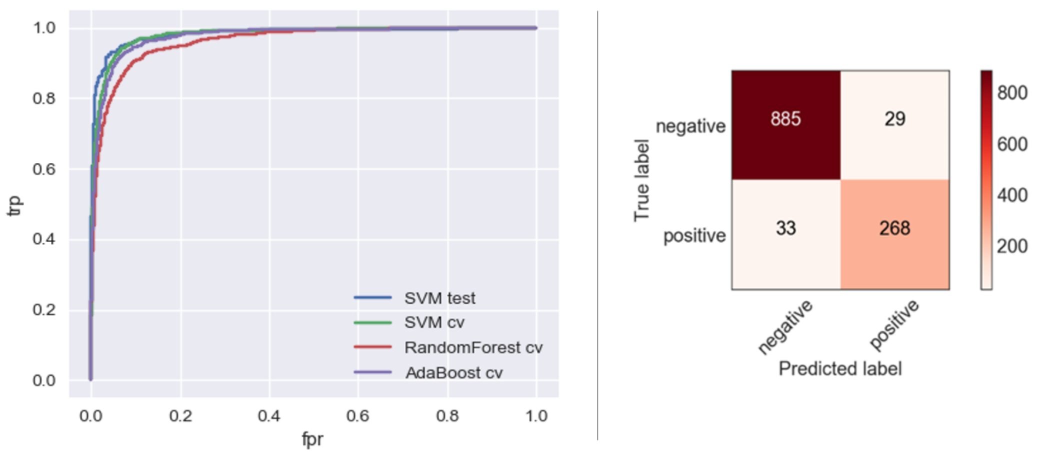


Figure 6 - Performance visualization of SVM

Left: A comparison of the ROC curves of all classifiers trained on WIKI data.

Right: Confusion matrix of WIKI trained SVM on test data

Pre-Processing

It was noticed that StanfordCoreNLP would raise warnings with some accented characters and non-encoded bytes. Furthermore, Doc2Vec and StanfordCoreNLP are both designed to work on UTF-8 encoded text and it seemed that there were some non-UTF-8 characters in the corpus which could be symbols or Chinese characters for example. They may appear in the articles as non-encoded bytes e.g. "\x00". The characters needed to be removed and the chosen method needed to be robust as it is expected to be used on a document stream and work on many different articles. To achieve this, an encoding mapping was used, which maps as many characters as possible into ASCII encoding. ASCII contains all of the most common characters in the English language such as the English alphabet, numbers and general punctuation. The first 128 characters of Unicode are the same as the ASCII encoding so it is a trivial task mapping from ASCII to UTF-8. Furthermore, there are generally well-known and accepted protocols for this task. Four different protocols (below table) were trialled: *Normalization Form Canonical Decomposition* (NFD), *Normalization Form Canonical Composition* (NFC), *Normalization Form Compatibility Decomposition* (NFKD) and *Normalization Form Compatibility Composition* (NFKC). NFKD was chosen as it gave the closest results to what was wanted.

Table 3 – Character Mapping Protocols

Original encoding	NFKD	NFKC	NFC	NFD
Ç, é, â, ê, î, ô, û, à, è, ù, ë, ï, ü, Α α, Β, β, Γ γ, Δ δ, Ε ε, Ζ ζ, Η η, Θ θ, Ι ι, Κ κ, Λ λ, Μ μ, Ν ν, Ξ ξ, Ο ο, Π π, Ρ ρ, Σ σ / ς, Τ τ, Υ υ, Φ φ, Χ χ, Ψ ψ, Ω ω, I, II, III, IV, V, VI, VII, VIII, IX, X, Д, µ, Ⅰ, Ⅱ,Ⅲ, Ⅳ, Ⅴ, Ⅵ, Ⅶ, Ⅷ, Ⅸ, Ⅹ, Ⅺ, Ⅻ , L, C, D, M, ↀ, ↁ, ↂ, Ↄↄ, ↅ, ↆ, ↇ, ↈ	C, e, a, e, i, o, u, a, e, u, e, i, u, , , , , , , , , , , , , , , , , , /, , , , , , , I, II, III, IV, V, VI, VII, VIII, IX, X, , ,I, II, III, IV, V, VI, VII, VIII, IX, X, XI, XII, L, C, D, M, , , , , , , , '	, ,\n , , , , , , , , / , , , , , , , I, II,\n III, IV, V, VI, VII, VIII, IX, X, , , I, II,III, IV, V, VI, VII, VIII, IX, X, XI, XII, L,\n C, D, M, , , , , , , , \n	, ,\n , , , , , , , , / , , , , , , , I, II,\n III, IV, V, VI, VII, VIII, IX, X, , , , ,, , , , , , , , , , ,\n , , , , , , , , , , \n	C, e, a, e, i, o, u, a, e, u, e, i, u, , , , , , , , , , , ,\n , , , , , , , , / , , , , , , , I, II,\n III, IV, V, VI, VII, VIII, IX, X, , , , ,, , , , , , , , , , ,\n , , , , , , , , , , \n

Another aspect of pre-processing is to remove duplicate content. It is apparent that some documents have duplicates and these may come from more than one publication. This could be because freelance journalists may sell a story to more than one publication. Furthermore, it was observed that when there was an important event such as an election or a large terror attack, some web-publications will have a running article, which may be updated often with new information as it happens. This creates another larger article in which the smaller article's content is included.

Comparing every document to every other document has $O(n^2)$ time complexity, so will take a long time with many documents. Additionally, it is not necessary since duplicate articles tend to be published at, approximately, the same time. Instead, the documents were sorted according to the extracted datetime objects, a 48 hours wide sliding window was passed across the corpus in steps of 24 hours. Documents were matched by simply checking if their strings were equal. If this was the case, the oldest document was retained. Though, it is possible that sometimes documents may differ by only a few characters or a single word and this process would miss these. However, without knowledge of what the additional characters or words are, any stricter filtering could remove useful information. Another possible approach would be to vectorize the documents and use a metric such as cosine similarity and treat pairs of documents that exceed an arbitrary threshold as duplicates. However, it was decided to opt for a process that has been named *Dechilding*. Whereby a *child* document is defined as a substring of another, *parent* document in the corpus. A test to see if a document is a child of another document is trivial in Python and just requires the syntax: *if string1 in string 2*. Both strings are tested against each other to see if either is the child of the other, then the parent is retained as this document will contain the most information.

After a survey of the documents in the corpus, it was noticed that there were occasionally documents which contained only a single word. It was first thought that these documents may have been the result of an error during scraping, but this was not the case as verified by checking with the Nexis database. It was decided to filter out these documents and any documents that contained less than 20 characters as it was observed that none would contain entities or useful information and this may help reduce noise and unnecessary processing.

Named Entity Recognition

Named entity recognition (NER) is the problem of identifying and classifying names in text. These may be persons such as *Albert Einstein*, locations such as *London* or organizations such as *Facebook*; the problem can also include other entity types. There are a few high-performing pre-trained classifiers, which are freely available. Using a pre-trained model is an acceptable solution for this project because training an effective model can, at the least, require a huge volume of supervised training data, and this would require too much time and human resources which are outside of the scope of this project. Furthermore, there are many pre-trained classifiers that are of high quality. Two of the most popular were selected which were suitable the project and initial trials were performed. For this, five documents with word counts that were approximately equal to the average word count of documents in the corpus (approx. 450 words and 2215 words in total) were selected, so that they appropriately represented the type of documents in the corpus, and they were manually compared to the outputs of the classifiers. There were some difficulties in doing this because StanfordNER and Spacy detect different types of entity. However, both of them detect persons, locations, and organizations but with some differences in the additional entity types. For instance, Spacy recognizes works of art and StanfordNER can recognize causes of death. These additional entities are of no use for this project, so they were filtered into a fourth category, which was labelled 'other'. Furthermore, the location labels between each classifier are handled somewhat differently. StanfordNER has separate entity types for countries, cities and locations whereas Spacy has a single label called GPE (countries, cities and states).

Spacy is a Python library which contains a number of NLP tools. It has three classifiers which are suitable for NER: en_core_web_sm, en_core_web_md and en_core_web_lg; denoting small, medium and large respectively. All three classifiers are Convolutional Neural Networks (CNNs) [18] which are essentially a multi-layered neural network which incorporate different types of layers but most notably non-fully-connected layers due to the use of a convolution filtering process. The CNNs in Spacy are pre-trained on data from the OntoNotes project [10]. This was a collaboration between some American Universities and Bolt Beranek and Newman Technologies. The goal of the project was to annotate a large corpus comprising various genres of text (news, conversational telephone speech, weblogs, usenet newsgroups, broadcast, talk shows). Furthermore, the training data contains some material from Agence France Press, which is a publication that is included in the corpus. Of the three classifier types, no notable differences in performance were observed, however, the large model was reported by the developers of Spacy to provide a small increase in accuracy over the other two as having been trained on a much larger training set. Therefore, this one was selected and is represented in the testing that is presented.

Stanford NER is the classifier that was used in the end process so a more detailed explanation will be given of how it works. Stanford NER is a software written in Java that provides a general implementation of (arbitrary order) linear chain Conditional Random Field (CRF) sequence models. Pioneered by Lafferty et al,. [46], a CRF is a probabilistic classifier. A good explanation of how a CRF works is provided by Sutton and McCallum [62] and first discusses the relationship between two simpler probabilistic classification models: Naive Bayes and Logistic Regression. Naive Bayes is generative, meaning that it is based on a model of the joint distribution $p(y, \mathbf{x})$ and thus makes the assumption that the input features, where $\mathbf{x} = \{x_1, x_2 \dots x_K\}$ a feature vector**,** are completely independent of each other.

Naive Bayes:

$$p(y, \mathbf{x}) = p(y) \prod_{k=1}^{K} p(x_k | y)$$

Logistic regression is discriminative, meaning that it is based on a model of the conditional distribution $p(y|\mathbf{x})$ and ergo models a probability over all x_i. This essentially means that it considers interdependence of the features.

Logistic Regression:

$$p(y| \mathbf{x}) = \frac{1}{Z(\mathbf{x})} exp \left\{ \sum_{k=1}^{K} \lambda_{k,} f_k(y, \mathbf{x}) \right\}$$

where $Z(x) = \sum_y \exp\{\lambda y + \sum_{j=1}^{k} \lambda_{y,j}\, x_j\}$ is a normalizing constant, and λ_y is a bias weight that acts like $\log p(y)$ in naive Bayes. The above equation is an altered form meant to resemble random fields. This has been achieved by defining a set of feature functions as $f_{y',j}(y, \boldsymbol{x}) = 1_{\{y'=y\}} x_j$ for the feature weights and $f_{y'}(y, \boldsymbol{x}) = 1_{\{y'=y\}}$ for the bias weights. f_k indexes each feature function $f_{y',j}$ and λ_y indexes its corresponding weight $\lambda_{y',j}$. These feature functions are only non-zero for a single class at a time.

Naive Bayes and Logistic Regression are fundamentally related in that a Naive Bayes classifier can be turned into a Logistic Regression classifier if we interpret it generatively:

$$p(y, \mathbf{x}) = \frac{\exp\{\sum_k \lambda_k\, f_k(y, \mathbf{x})\}}{\sum_{\tilde{y}, \tilde{x}} \exp\{\sum_k \lambda_k\, f_k(y, \mathbf{x})\}}$$

and vice versa if the logistic regression model is trained to maximize $p(y, \mathbf{x})$. The main difference between generative and discriminative models is that the conditional distribution of a discriminative model does not include a model of $p(\mathbf{x})$ which is not necessary for classification. This is advantageous because, as with the NER problem, the features of $\mathbf{x}$ can be highly interdependent.

An HMM models a sequence of observations $\boldsymbol{X} = \{x_1, x_2 \dots x_t\}_{t=1}^{T}$ and assuming a set of underlying states $\boldsymbol{Y} = \{y_1, y_2 \dots y_t\}_{t=1}^{T}$. It addresses the problem of interdependence by making the assumptions that each y_t is dependent exclusively on the preceding state y_{t-1} and independent of all other states. Secondly, it assumes that x_t is dependent exclusively on the present state y_t. However, there are obvious issues with doing this

Sutton and McCallum highlight a parallel between the relationship between Naive Bayes and Logistic Regression to the relationship between Hidden Markov Model (HMM) and a CRF. In this case HMM would be analogous to Naive Bayes adapted for operating on sequences and the same relationship inferred between Logistic Regression and CRF.

The NER problem is concerned with sequences of words in which it would be advantageous to consider, not only the name of an entity, but also its context e.g. surrounding words, punctuation and even the entire document if possible. CRFs model the conditional $p(\mathbf{y}|\mathbf{x}) = p(y_1, y_2 \dots y_T|x_1, x_2 \dots x_T)$ and hence make independence assumptions among $\mathbf{y}$, but not among $\mathbf{x}$. Essentially, CRFs are a discriminant probabilistic classifier which model the conditional probability $p(\mathbf{y}|\mathbf{x}) = p(y_1, y_2 \dots y_T|x_1, x_2 \dots x_T)$ where $\mathbf{x}$ is a sequence of words, and $\mathbf{y}$ is a set of hidden states. CRFs assume independence among $\mathbf{y}$, but not $\mathbf{x}$ and so assume independence of hidden states but consider the context of the entire document.

It can be seen in the results table below that StanfordNER outperforms Spacy for this task. Both classifiers tend to identify more entities than exist in the text. However, this problem is more profound with Spacy. Both classifiers have some difficulty distinguishing between persons and organizations. Locations tend to be the easiest of the three to detect.

Table 4 – Results from Trialling NER Classifiers

	Stanford				Spacy				
	Acc.	Precision	Recall	f1	Acc.	Precision	Recall	f1	n
Entity		0.87	1.00	0.93		0.83	1.00	0.91	352
Non-entity		1.00	0.98	0.99		1.00	0.97	0.98	2215
People	0.86	0.91	0.76	0.83	0.75	0.82	0.60	0.69	89
Locations		0.97	1.00	0.98		0.70	1.00	0.82	32
Organizations		0.69	0.75	0.72		0.55	0.54	0.55	57
Other		0.74	1.00	0.85		0.68	1.00	0.81	174
						Total word count			2567

Considering the above results, it was decided that StanfordNER was the best classifier to use for the task. StanfordNER returns an output whereby each document is represented by a (Python) list of lists, where each nested list represents a sentence and contains the entities that occurred in that sentence in the document. An entity is represented by an object with *word, words* and *tag,* attributes that store the entity's single-string text representation, a list of strings of the words in that string and the entity type i.e. person, location or organization, respectively. It should be noted that only the main body of text from each document has been used and the titles, by-lines (i.e. authors) and datelines have been ignored. The titles tended to be recognized as a single entity because each word is normally capitalized. Authors and datelines could have also added noise, for example it seemed likely that an author may be included in the corpus multiple times and could theoretically become a node with high centrality but be unimportant in detecting terror.

Named Entity Disambiguation

Named Entity disambiguation (NED) is itself quite a challenge. There are many options for how this may be approached. NED is often synonymous with *Entity Linking*. This is the process of linking the mentions of an entity to an entity in a database. The way this is performed is that mentions of an entity are queried in a database, sometimes over the internet, such as Wikipedia [39]. The query returns search results and the most relevant, often the top result, is taken to be the entity. However, the primary concern was with the disambiguation part of the process and it was only necessary to match occurrences of entities within the corpus. This means that the only concern was with resolving the string representations of a single entity to the same string. For example, the same entity may be referred to by more than one name such as *UK* for *United Kingdom* or *USA* for *United States*. Often persons may be referred to by their surname e.g. *Trump* for *Donald Trump*. The aim is to resolve all of the different names that may be used to refer to that entity, to a single name. This reduces ambiguity in the networks and helps make the networks more interpretable.

Initially some cleaning of the entity words is performed. It was noticed that sometimes there may be hyphenated words attached to entities such as *-born* e.g. *British-born* or *US-backed* etc… Some simple functions were implemented to rectify the errors that were noticed.

Some of the functions exploit some simple, often followed patterns, of formal writing which are often observed in news media. Furthermore, the process makes use of the entity type tags i.e. *person, location* and *organization*, assigned by the NER process, and treats each type differently in some situations. StanfordNER outputs multiple entity types that are what could be considered locations i.e. *countries, cities and locations*. Furthermore, entities tagged with *nationality* were considered as a reference to a nation and therefore *location*. These entity tags are all resolved to a single tag of *location*. Each entity was given a *resolved* attribute that is a boolean that signifies if an entity is fully resolved. This is to stop further detrimental changes being made to entities that do not need any disambiguation.

A simple way to disambiguate locations is to cross reference them with a local database. For this, data from a couple of different sources was used, *Simplemaps* [15] which has an open source database of approximately 13K cities, accumulated from organizations such as the *National Geospatial Intelligence Agency* (United States) and *NASA*. Python library known as *pycountry* [16], was also used, which contains the ISO databases for 639-3 (languages), 3166 (countries), 3166-3 (deleted countries), 3166-2 (subdivisions of countries), 4217 (currencies) and 15924 (scripts). The country names (not including acronyms) are put through the pre-processing step to ensure any accents are resolved. They are treated the same as to increase the chances that they will match.

Additionally, a set of persons of interest (POI) obtained from the GTD was used. This was created by scraping the descriptions of each terror event in 2017, pre-processing them and then using StanfordNER. Furthermore, a database of known perpetrator groups (KPG) was created from the list of all know perpetrator groups available in the GTD. This set is not limited to events in 2017 but contains the names and acronyms of all know terrorist organizations. The KPG (excluding acronyms) were put through the pre-processing step.

A single-word (entity) where its length is greater than 1 and lower than or equal to 4, and is all uppercase, is assumed to be an acronym for a longer multi-word entity. The process iterates through the preceding entities in the document and matches the acronym's letters to the starting letters of each entity's *words* attribute (explained in previous section), ignoring stopwords such as *of*, and *the*. Acronyms that have matches are resolved to the matching entity and then considered resolved. After this, any unresolved acronyms are queried in the ISO database of *pycountry*, and if no match, are queried in the KPG database. Any remaining after this step are not further resolved.

All location entities are checked against the pycountry database which has a fuzzy lookup function and, if not resolved, the Simplemaps database.

An entity such as a person, may frequently be referred to by only their surname but often only after their full name has been referenced earlier in the document. For example, an article that discusses *Donald Trump* may use the full name *Donald Trump* near the beginning of the article to indicate that all later references to *Trump* are referring to Donald Trump. Therefore, a function is used that, for any single-word-entity, (ent1) iterates through the document in reverse and finds the first more-than-one-word-entity (ent2) that has a start word or end word that matches that of ent1. If a match is found, ent1 is made equal to ent2. This process is constrained to the scope of preceding entities within the same document as otherwise entities may be matched to entities to which they have no relation. Any unresolved person entities are then checked against the POI database.

Network Analysis

In this section the extracted entities are analysed by network analysis to identify patterns and assess the potential for event detection.

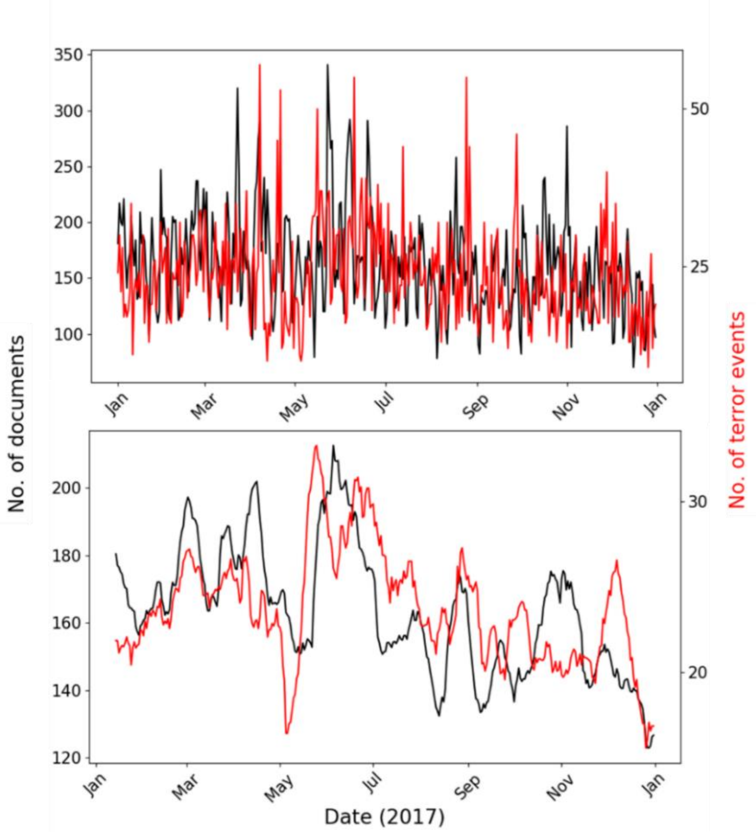


Figure 7 - Timeseries comparison of document count to number of terror events
Top: Raw counts. Bottom: 2 week rolling average

In the above figure the number of documents per day is compared, after filtering by SVM, to a count of the number of terror events in the GTD per day. It can be seen that both timeseries are volatile and it is difficult make comparisons, but it does appear there are some similar trends. Once some of the volatility is smoothed out by applying a rolling average, the trends are more apparent. This is a good indication, although, it should be noted that it cannot be said that it shows a causal relationship, but if there was no similar trends it could more strongly suggest that the SVM filter did a poor job. Furthermore, if there is a casual relationship it cannot be said that it is due to terror events. The trends are over large timescales so could be related to seasonal things (e.g. a reduction near seasonal holidays like Christmas or winter) and it can be seen in the top plot that there are many instances where there are many terror events that do not correspond to many documents and vice versa. Nevertheless, it was expected that the GTD and news media would not necessarily align perfectly, as it is likely that some terror events may not exist in the sample of documents that were downloaded and it is possible that unrelated documents have made it through the processing thus far.

Co-occurrence rules

The networks that have been considered are undirected. Some experiments with different co-occurrence rules were performed. One of these was to consider the co-occurrence of entities within the same sentence, another considered only the co-occurrence of entities within the same document and the last considered both sentence and document co-occurrence. Furthermore, two different ways to quantify these co-occurrence rules were trialled. These were *significance* as described in [52] and a simple *raw count* method. The significance S of an entity v in a text x is given by:

$$S_{x(v)} = \frac{tf(v,x)}{\sum_{v' \in V} tf(v',x)}$$

where tf is the *term frequency* or raw count of entity v in text x and the denominator is the raw count of all entities that occur in x. An edge weight w between entities i and j, in the scope of a single document d, is given by:

$$w_d(i,j) = \begin{cases} \sum_{r \in d} \left(S_r(i) + S_r(j)\right) & if\ i,j \in V_d \\ 0 & otherwise \end{cases}$$

where r is a sentence in d and i and j co-occur in the set of entities V_d, which occur in d. Similarly, if we are to consider the significance of co-occurrence exclusively in the document:

$$w_d(i,j) = \begin{cases} S_d(i) + S_d(j) & if\ i,j \in V_d \\ 0 & otherwise \end{cases}$$

and then to consider both:

$$w_d(i,j) = \begin{cases} S_d(i) + S_d(j) + \sum_{r \in d} \left(S_r(i) + S_r(j)\right) & if\ i,j \in V_d \\ 0 & otherwise \end{cases}$$

Essentially, we have now constructed a graph for each document. To create a graph within a given time window, the edges are the summation of corresponding edges in the document graphs within that time window:

$$W(i,j) = \sum_{d \in D_W} w_d(i,j)$$

The other co-occurrence rule we experimented with gives an edge weight the raw count of co-occurrences between entities:

$$w_x(i,j) = \begin{cases} tf(i,x) \times tf(j,x) & if\ i,j \in V_x \\ 0 & otherwise \end{cases}$$

where x may be a sentences or documents or both. Hence, in the case of one document and if we are only considering document co-occurrence: if entity i and j both occur once in the document $w_d(i,j) = 1$. If entity i occurs once and j occurs twice (or vice versa) $w_d(i,j) = 2$ and if either i or j do not occur $w_d(i,j) = 0$.

Centrality Metrics

Degree Centrality

Here a brief explanation is provided of some of the centrality metrics that were used as to aid further discussions. It is encouraged to read the reference if more clarification is required [55].

A centrality metric essentially quantifies the relative importance of nodes (entities) in the network. The most simple of these is *degree* centrality, whereby, if we consider a single node i, its degree is the number of edges which connect it to other nodes. It is also possible to consider the weights of the edges, hence, the *weighted degree* of i is the sum of the edges which connect i to other nodes:

$$M_i = \sum_{i \in J} W(i,j)$$

where M_i is the weighted degree of i and j is an entity in the set of entities J connected by a weighted edge to i.

Eigenvector Centrality
Another extension on the concept of degree centrality is *eigenvector centrality* [24]. The idea of eigenvector centrality is to consider, not only the adjacent nodes, but the nodes adjacent to the adjacent nodes. Basically, a node will be more important if it has adjacency to other nodes which are important. Let us consider an adjacency matrix A which mathematically encodes the entire network. Here each row represents a node i and each column represents a node j. An edge in a simple undirected and unweighted network may be denoted:

$$A_{i,j} = \begin{cases} 1 & \text{if edge between } i \text{ and } j \\ 0 & \text{otherwise} \end{cases}$$

The matrix will be populated with ones and zeros with the main diagonal being zeros because when $i = j$ there will be no edge. Furthermore, in the case of an undirected network the matrix will be symmetric as $A_{i,j} = A_{j,i}$. In the case of a weighted network the $A_{i,j}$ is the edge weight between i and j. Eigenvector centrality is yielded:

$$M_i = k_1^{-1} \sum_j A_{i,j} M_j$$

where $k_1$is the largest eigenvector of A.

PageRank
This centrality metric is well known for being incorporated in Google's search engine and has had success in ranking websites. PageRank [30] is actually intended for use on directed networks but highly related to degree in the case of undirected networks [36]. It is meant to improve on a weakness of another centrality metric known as *Katz* centrality which basically works by giving all nodes an equal amount of importance which they then distributes evenly to each of their neighbours. The weakness pertains to: if a node with high centrality links many others then all those others get high centrality. PageRank attempts to dilute the *free* centrality that a node gives out as to not disproportionately over-rank nodes. The equation is given below:

$$M_i = \propto \sum_j A_{i,j} \frac{M_j}{k_j} + \beta_i$$

where $\propto$ is a hyperparameter which has been left as 0.85, β_i is the *free* importance which a node distributes and has been left as 1.0, the k_j term is normally denoted k_j^{out} and is the out-degree of j but in the case of undirected networks it is just the degree.

An experiment was performed by which networks were implemented using different combinations of the aforementioned centrality metrics and co-occurrence rules and the top 40 scoring entities were observed. The results are tabulated in tables 5 to 16. Overall, the sets of entities that were observed contained a high proportion on those that one would expect to have collected with a method designed to detect terrorism e.g. terrorist organizations and locations which have frequent terror events (according to the GTD). Additionally, there are very few entities that have been labelled with the incorrect type e.g. persons have been identified as persons and locations as locations. However, it can also be seen that there is also substantial representation of world leaders such as *Donald Trump* or *Vladimir Putin*. It seems that world leaders could be often mentioned in news articles reporting on terror events. Though, it remains to be seen if they are useful indicators of terror events. For example, Donald Trump is often the person entity with the highest centrality but it seems unlikely that is because he is highly related to terror events but seems more probable that it is because he occurs in a lot of articles. Donald Trump could be a good indication that there is a lot of noise in the system. Furthermore, it is also a possibility that the high representation of world leaders could be due to articles which are not related or weakly related to terror events making it through the filtering process.

Another similar observation is that there is also quite a substantial representation of the press in many of the top 40s e.g. reporter names or publication names. It is apparent that these must have existed in the main body of texts as by-lines and authors were not used in the NER process. The names of the publications in the corpus had also been filtered out. Most notable, there is an entity named *Spot Development* which has been identified as an organization and is ranked highly by all centrality metrics and co-occurrence rules. This entity is actually a title which occurs often in the articles by *Associated Press International* and denotes the type of article.

It can be seen that out of the three entity types that have been considered, that *locations* have a tendency to be highly represented amongst the nodes with the highest centrality. This remains true for all centrality metrics that have been used and whether sentence or document co-occurrence is considered (tables 5 to 7). The location node with the highest centrality is the *United States*, which often has a score approximately double that of the next highest node. This is highly disproportionate with the number of terror attacks that have occurred in the United States in 2017 as seen in the GTD. In sight of this, the location entities were removed different combinations of *person* and *organization* entity types were investigated (tables 8 to 16).

Interestingly, when the eigenvector centrality metric was used with the sentence co-occurrence rule on person entities, the top 50 were very different to those that had been previously seen (table 12). The entities returned were highly related to some of the events in the GTD. This was also apparent in another experiment that was performed whereby the eigenvector centrality versus degree for each entity was plotted. Some of the results can be seen in figure 9. It was apparent that for every combination there was a positive correlation e.g. both metrics yielded similar results, except for unweighted sentence significance, unweighted count and weighted sentence count for person entities. With weighted sentence count there was very low variance in eigenvector centrality compared to degree. This suggests that the persons of interest tend to be adjacent to other nodes with many edges.

It can be seen in the below figure that when considering only sentence co-occurrence, there are few nodes with high centrality and a vast majority have relatively low centrality. When document co-occurrence is considered, a large portion of the nodes gain centrality. The distribution is less heavily weighted at the low end of centrality. When considering both sentence and document co-occurrence, it can be seen that the distribution most closely resembles document co-occurrence which suggests that this is the component that makes up the majority of the edges.

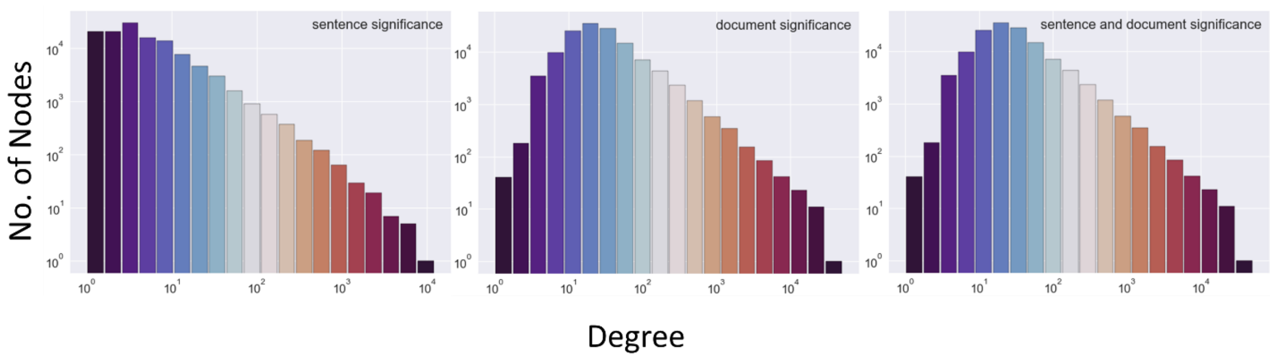


Figure 8 – Distribution of Degree Centrality (log scale)
Left: Sentence co-occurrence. Centre: Document co-occurrence. Right: Sentence and Document co-occurrence.

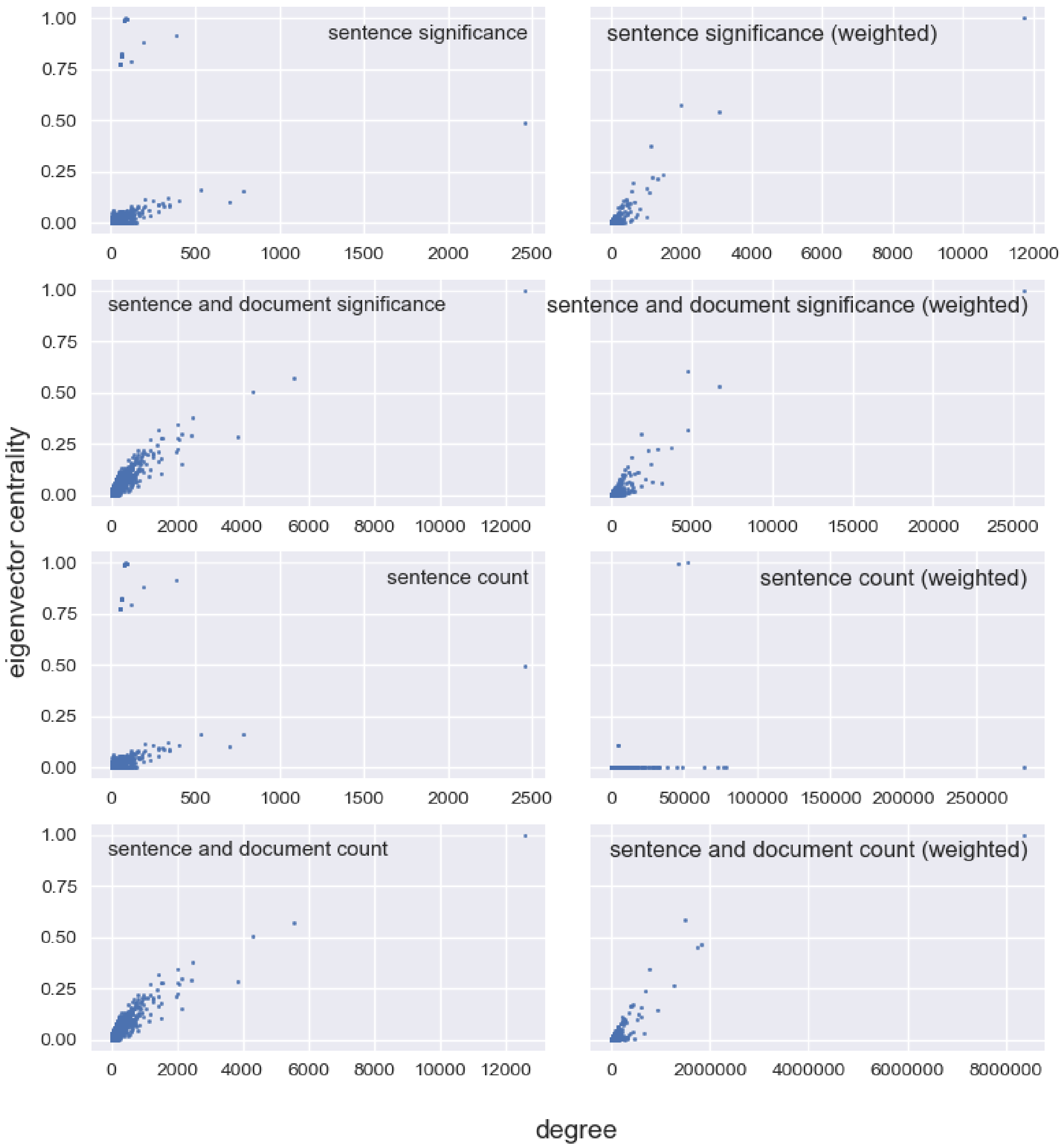


Figure 9 - Eigenvector Centrality versus Degree Centrality

Table 5 - Top 50 Entities of All Types by Degree Centrality

Unweighted				Weighted (Significance)			
Sentence		Sentence and document		Sentence		Sentence and document	
United States	12119	United States	65160	United States	50438	United States	110503
Syria	6777	Syria	30432	Syria	33284	Donald Trump	49503
Russia	5841	Donald Trump	30133	Russia	26182	Syria	46080
Donald Trump	5206	Russia	20885	Iraq	25731	ISIL	34681
Pakistan	5180	Afghanistan	20668	Donald Trump	25279	Afghanistan	32007
Afghanistan	5072	Pakistan	16806	Afghanistan	22498	Russia	31400
Taliban	5029	Iraq	15949	Spot Development	21662	Iraq	25609
ISIL	4352	Taliban	15320	ISIL	21620	Pakistan	25291
Iraq	4334	ISIL	15159	Pakistan	19698	Taliban	23852
Iran	3876	Iran	14442	Iran	18593	Iran	22280
Turkey	3583	Turkey	13498	Washington	18215	Turkey	20519
Washington	3187	Israel	10755	Turkey	17904	United Nations	18743
Moscow	3059	Kabul	10280	United Nations	17639	Kabul	17812
Israel	2907	Saudi Arabia	10273	Moscow	16031	Saudi Arabia	17778
United Nations	2724	United Nations	9572	France	15059	NATO	16439
Kabul	2675	Moscow	7711	Europe	14753	Israel	16144
London	2601	Washington	7634	Taliban	14304	UAE	13175
Egypt	2586	NATO	7139	Middle East	13921	Moscow	13087
Saudi Arabia	2564	Egypt	7046	Saudi Arabia	13875	Washington	12948
France	2463	UAE	6946	London	13251	Spot Development	12035
Ukraine	2218	Mosul	6807	United Kingdom	13206	Vladimir Putin	11675
United Kingdom	2194	Vladimir Putin	6565	Israel	13196	Egypt	11324
India	2118	France	6147	Arab	12964	Bashar Al Assad	11276
Paris	2090	White House	5785	Kabul	12741	Mosul	10994
Arab	2045	Jerusalem	5622	America	12196	Pashto	10834
UAE	2036	Yemen	5515	Paris	12069	France	10455
Daesh	2031	Bashar Al Assad	5443	Egypt	11798	White House	10333
Jerusalem	2012	Pashto	5440	Vladimir Putin	11649	Middle East	9024
Mosul	1997	Ukraine	5260	European Union	11587	Kurdistan WP	8875
Libya	1888	Somalia	5161	Germany	11556	Yemen	8823
Cairo	1860	London	5158	White House	11172	London	8789
Vladimir Putin	1854	India	5116	Interior Ministry	11133	Arab	8743
New York	1827	Paris	5037	Britain	10869	United Kingdom	8675
Somalia	1822	Arab	4864	India	10353	Somalia	8644
Interior ministry	1815	Middle East	4532	Pashto	10078	European Union	8564
Middle East	1730	European Union	4454	NATO	10005	Paris	8552
Germany	1719	Libya	4449	New York	9886	Ukraine	8539
Europe	1668	United Kingdom	4374	Ukraine	9835	Jerusalem	8523
White House	1649	Kurdistan WP	4297	Mosul	9678	Dari	7858
European Union	1639	Qatar	4188	Libya	9651	Recep T Erdogan	7824

Table 6 - Top 50 Entities of All Types by Eigenvector Centrality

Unweighted				Weighted (Significance)			
Sentence		Sentence and document		Sentence		Sentence and document	
United States	1	United States	1	United States	1	United States	1
Russia	0.681	Syria	0.809	Donald Trump	0.719	Donald Trump	0.712
Syria	0.662	Donald Trump	0.743	Syria	0.535	Syria	0.523
Donald Trump	0.603	Russia	0.707	Afghanistan	0.461	Afghanistan	0.424
ISIL	0.571	Iraq	0.698	Russia	0.420	Russia	0.399
Afghanistan	0.554	ISIL	0.629	Iraq	0.326	Iraq	0.324
Turkey	0.552	Washington	0.617	Pakistan	0.297	ISIL	0.317
Pakistan	0.522	Iran	0.593	Iran	0.255	Pakistan	0.267
Iraq	0.520	Turkey	0.579	ISIL	0.249	Iran	0.252
Washington	0.498	United Nations	0.556	Turkey	0.239	Turkey	0.233
Saudi Arabia	0.492	Afghanistan	0.550	Washington	0.193	NATO	0.221
Iran	0.487	Moscow	0.540	Taliban	0.187	Washington	0.216
Egypt	0.465	Middle East	0.539	NATO	0.183	Taliban	0.193
France	0.443	Europe	0.534	Israel	0.165	White House	0.169
London	0.437	France	0.528	White House	0.153	Bashar Al Assad	0.165
United Kingdom	0.431	Spot Development	0.525	Bashar Al Assad	0.143	Israel	0.161
Moscow	0.428	Saudi Arabia	0.508	Vladimir Putin	0.135	Vladimir Putin	0.144
United Nations	0.428	United Kingdom	0.499	Barack Obama	0.132	Barack Obama	0.143
Israel	0.414	America	0.488	Moscow	0.127	Moscow	0.142
UAE	0.413	White House	0.487	Saudi Arabia	0.125	Saudi Arabia	0.142
Germany	0.408	Israel	0.482	United Nations	0.108	United Nations	0.135
Paris	0.403	Vladimir Putin	0.480	Jerusalem	0.105	Kabul	0.126
New York	0.399	Germany	0.477	Kabul	0.092	Middle East	0.107
White House	0.398	London	0.476	Mosul	0.090	Mosul	0.101
Libya	0.394	Pakistan	0.472	Middle East	0.090	Jerusalem	0.100
Kabul	0.382	Arab	0.468	Yemen	0.088	Yemen	0.094
Cairo	0.371	Paris	0.460	Rex Tillerson	0.084	Kurdistan WP	0.089
Jordan	0.370	European Union	0.457	America	0.073	America	0.089
Middle East	0.364	Britain	0.446	India	0.073	Rex Tillerson	0.085
Arab	0.359	Egypt	0.441	Kurdistan WP	0.072	Recep T Erdogan	0.083
Istanbul	0.345	NATO	0.428	Recep T Erdogan	0.070	Pentagon	0.079
Vladimir Putin	0.339	Barack Obama	0.426	Pentagon	0.070	Arab	0.079
Riyadh	0.337	China	0.422	Arab	0.068	European Union	0.074
America	0.336	Libya	0.420	European Union	0.066	Spot Development	0.072
European Union	0.334	New York	0.418	Raqqa	0.065	UAE	0.072
Europe	0.333	Yemen	0.418	Bashar Assad	0.064	United Kingdom	0.068
Jerusalem	0.331	West	0.417	Somalia	0.062	Raqqa	0.068
Brussels	0.322	Foreign Ministry	0.403	Ukraine	0.060	India	0.067
Britain	0.314	UAE	0.391	China	0.059	France	0.066
Qatar	0.312	Mosul	0.385	United Kingdom	0.059	Somalia	0.065

Table 7 - Top 50 Entities of All Types by PageRank Centrality

Unweighted				Weighted (Significance)			
Sentence		Sentence and document		Sentence		Sentence and document	
United States	8.1E-03	United States	5.9E-03	United States	2.5E-02	United States	2.4E-02
Syria	4.4E-03	Syria	3.8E-03	Syria	1.1E-02	Donald Trump	1.1E-02
Taliban	3.9E-03	Iraq	2.9E-03	Donald Trump	1.1E-02	Syria	9.7E-03
Russia	3.6E-03	Spot Development	2.9E-03	Afghanistan	8.3E-03	ISIL	8.1E-03
Afghanistan	3.5E-03	Russia	2.8E-03	Russia	8.0E-03	Afghanistan	7.2E-03
Pakistan	3.4E-03	Afghanistan	2.8E-03	Pakistan	7.2E-03	Russia	6.8E-03
Donald Trump	3.3E-03	Donald Trump	2.6E-03	Taliban	7.0E-03	Pakistan	6.2E-03
ISIL	2.8E-03	Pakistan	2.4E-03	ISIL	6.1E-03	Taliban	5.7E-03
Iraq	2.8E-03	ISIL	2.4E-03	Iraq	6.0E-03	Iraq	5.6E-03
Iran	2.5E-03	Iran	2.1E-03	Iran	5.4E-03	Iran	4.8E-03
Turkey	2.2E-03	United Nations	2.0E-03	Turkey	4.9E-03	Turkey	4.3E-03
Kabul	1.9E-03	Turkey	2.0E-03	Kabul	4.2E-03	United Nations	4.3E-03
Moscow	1.9E-03	Taliban	1.9E-03	Israel	4.0E-03	Kabul	4.1E-03
Washington	1.8E-03	Washington	1.9E-03	United Nations	3.8E-03	Saudi Arabia	3.8E-03
Israel	1.8E-03	Moscow	1.7E-03	Saudi Arabia	3.7E-03	NATO	3.7E-03
United Nations	1.7E-03	Kabul	1.7E-03	Moscow	3.1E-03	Israel	3.4E-03
London	1.7E-03	France	1.6E-03	Washington	2.9E-03	Spot Development	3.2E-03
France	1.6E-03	Europe	1.5E-03	Mosul	2.8E-03	UAE	3.0E-03
Egypt	1.6E-03	Saudi Arabia	1.5E-03	Egypt	2.8E-03	Moscow	2.9E-03
Mosul	1.5E-03	Pashto	1.5E-03	UAE	2.7E-03	Washington	2.8E-03
Saudi Arabia	1.5E-03	Israel	1.5E-03	NATO	2.7E-03	Vladimir Putin	2.6E-03
Ukraine	1.4E-03	Arab	1.4E-03	France	2.6E-03	Egypt	2.6E-03
India	1.3E-03	Middle East	1.4E-03	Vladimir Putin	2.5E-03	France	2.5E-03
Paris	1.3E-03	London	1.4E-03	Somalia	2.5E-03	Mosul	2.5E-03
United Kingdom	1.3E-03	United Kingdom	1.4E-03	London	2.3E-03	Pashto	2.5E-03
Daesh	1.3E-03	Egypt	1.3E-03	India	2.3E-03	Somalia	2.3E-03
Somalia	1.3E-03	Interior Ministry	1.3E-03	Ukraine	2.2E-03	Bashar Al Assad	2.3E-03
Arab	1.2E-03	Paris	1.3E-03	Paris	2.2E-03	London	2.2E-03
Jerusalem	1.2E-03	America	1.2E-03	Jerusalem	2.1E-03	White House	2.2E-03
Interior Ministry	1.2E-03	European Union	1.2E-03	White House	2.1E-03	United Kingdom	2.1E-03
UAE	1.2E-03	India	1.2E-03	Yemen	2.0E-03	Paris	2.1E-03
Cairo	1.1E-03	Germany	1.2E-03	Pashto	1.9E-03	India	2.0E-03
Vladimir Putin	1.1E-03	Mosul	1.1E-03	United Kingdom	1.9E-03	European Union	2.0E-03
Libya	1.1E-03	Daesh	1.1E-03	Bashar Al Assad	1.9E-03	Kurdistan WP	2.0E-03
New York	1.1E-03	Vladimir Putin	1.1E-03	Arab	1.8E-03	Middle East	2.0E-03
Germany	1.1E-03	Britain	1.1E-03	Libya	1.8E-03	Ukraine	2.0E-03
Europe	1.0E-03	Dari	1.1E-03	European Union	1.8E-03	Arab	1.9E-03
Middle East	9.9E-04	Somalia	1.1E-03	Middle East	1.7E-03	Yemen	1.9E-03
America	9.9E-04	White House	1.1E-03	Kurdistan WP	1.7E-03	Al Shabab	1.8E-03
AIP	9.8E-04	Ukraine	1.1E-03	Al Shabab	1.7E-03	Jerusalem	1.8E-03

Table 8 - Top 50 Organizations by Degree Centrality

Unweighted				Weighted (Significance)			
Sentence		Sentence and document		Sentence		Sentence and document	
ISIL	1352	ISIL	7298	Taliban	5040	ISIL	12726
Taliban	1236	United Nations	5853	ISIL	3769	Taliban	10961
United Nations	959	Spot Development	5738	United Nations	2427	United Nations	8764
Interior Ministry	639	Taliban	3994	NATO	1361	Spot Development	7347
NATO	529	Interior Ministry	3677	Kurdistan WP	1342	NATO	5914
Foreign Ministry	401	NATO	3626	Hamas	1071	Kurdistan WP	3611
FBI	397	Foreign Ministry	2840	Voice Of Jihad	889	Interior Ministry	3529
Hamas	349	FBI	2395	FBI	878	Pentagon	2364
Kurdistan WP	342	European Union	2326	Interior Ministry	878	FBI	2352
Pentagon	312	Congress	2220	Pentagon	828	Hamas	2332
AIP	310	UN Security Council	2090	AIP	801	Foreign Ministry	2320
UN Security Council	287	Pentagon	2023	SDF	789	Al Shabab	2290
Defence Ministry	282	Senate	1965	Al Shabab	696	Syrian Observatory For Human Rights	2276
Congress	275	Supreme Court	1777	Security Service Of Ukraine	678	UN Security Council	2219
Interfax Ukraine	270	CIA	1552	Interfax Ukraine	598	European Union	1822
Supreme Court	257	Defence Ministry	1529	HTS	589	Hezbollah	1817
Hezbollah	254	State Dept.	1474	Foreign Ministry	587	SDF	1742
Senate	252	Hezbollah	1455	Islamic Emirate	577	Voice Of Jihad	1731
Al Shabab	251	Al Qaida	1382	Hezbollah	575	Congress	1521
CIA	239	Hamas	1362	SBU	555	AIP	1488
European Union	228	Islamic Salvation Front	1293	Al Qaida	534	Security Service Of Ukraine	1479
Boko Haram	226	Kurdistan WP	1253	UN Security Council	531	Luhansk People's Republic	1393
Security Service Of Ukraine	204	Security Council	1248	Luhansk People's Republic	500	Al Qaida	1321
Express Tribune	200	Us State Dept.	1198	European Union	494	State Dept.	1300
Ministry Of Foreign Affairs	190	Reuters	1152	Congress	471	Senate	1284
Al Qaida	188	Ministry Of Foreign Affairs	1136	Boko Haram	459	Interfax Ukraine	1270
SDF	188	Al Shabab	1121	IRGC	458	Boko Haram	1260
Hurriyet	188	Foreign Office	1096	Senate	430	Islamic Salvation Front	1199
IRGC	185	PTI	1092	Hurriyet	429	Supreme Court	1163
Air Force	185	Air Force	1069	CIA	423	Defence Ministry	1158
SBU	184	AP	1030	IRGC	382	Hurriyet	1112
State Dept.	184	National Assembly	933	Palestinian Authority	379	Security Council	1095
Islamic Salvation Front	182	Cctv	914	Defence Ministry	364	CIA	1086
Pakistan Army	175	Muslim Brotherhood	914	Security Council	363	National Directorate Of Security	1056
Tass	169	State Duma	912	Libyan National Army	354	Muslim Brotherhood	1005
Army	163	Constitution	910	FSB	353	HTS	968
Ria Novosti	159	FSB	909	Supreme Court	353	SBU	947
IRGC	156	Justice Dept.	862	State Dept.	348	Palestinian Authority	936
PTI	153	Nation	859	GNA	340	IRGC	934
FSB	151	Army	826	Pakistan Army	337	Islamic Emirate	899

Table 9 - Top 50 Organizations by Eigenvector Centrality

Unweighted				Weighted (Significance)			
Sentence		Sentence and document		Sentence		Sentence and document	
ISIL	1	ISIL	1	Taliban	1	Taliban	1
United Nations	0.760	United Nations	0.926	Voice Of Jihad	0.744	NATO	0.625
Taliban	0.716	Spot Development	0.740	Islamic Emirate	0.384	Voice Of Jihad	0.570
NATO	0.491	NATO	0.708	AIP	0.371	ISIL	0.540
Interior Ministry	0.490	Foreign Ministry	0.635	NATO	0.322	Spot Development	0.410
Foreign Ministry	0.437	Interior Ministry	0.611	ISIL	0.232	United Nations	0.408
Pentagon	0.372	Taliban	0.555	Haqqani Network	0.107	AIP	0.315
UN Security Council	0.345	European Union	0.547	National Directorate Of Security	0.107	Islamic Emirate	0.264
Defence Ministry	0.295	UN Security Council	0.542	Daily Afghanistan	0.087	Interior Ministry	0.216
Congress	0.279	Pentagon	0.517	Maydan Wardag Province	0.086	Pentagon	0.210
Hezbollah	0.272	Congress	0.476	United Nations	0.085	Kurdistan WP	0.186
Hamas	0.272	Senate	0.466	Al Qaida	0.083	National Directorate Of Security	0.162
CIA	0.262	FBI	0.456	Pentagon	0.074	UN Security Council	0.162
Kurdistan WP	0.257	CIA	0.439	Nangarhar Province	0.074	Syrian Observatory For Human Rights	0.132
Senate	0.256	State Dept.	0.439	High Peace Council	0.066	Hurriyet	0.130
FBI	0.255	Al Qaida	0.414	Afghan National Army	0.062	SDF	0.125
European Union	0.251	Supreme Court	0.400	Interior Ministry	0.054	Al Qaida	0.121
Al Qaida	0.241	Us State Dept.	0.399	National Union Of Journalists Of Afghanistan	0.053	European Union	0.120
State Dept.	0.241	Security Council	0.398	Shahin Army Corps	0.045	Foreign Ministry	0.110
Us State Dept.	0.227	Hezbollah	0.384	Sar E Pol Province	0.043	Security Council	0.104
Supreme Court	0.219	Defence Ministry	0.376	Islamic State Of Iraq	0.043	Islamic State Of Iraq	0.092
Air Force	0.216	Reuters	0.341	American University Of Afghanistan	0.041	Haqqani Network	0.092
Islamic Salvation Front	0.212	Ministry Of Foreign Affairs	0.338	Konar Province	0.041	FBI	0.085
AIP	0.208	Hamas	0.318	Farah Province	0.040	Hezbollah	0.085
SDF	0.203	AP	0.316	Kurdistan WP	0.039	Daily Afghanistan	0.085
Ministry Of Foreign Affairs	0.201	Islamic Salvation Front	0.313	Hurriyet	0.039	State Dept.	0.080
Security Council	0.201	Air Force	0.308	Jamaat Ul Ahrar	0.038	Afghan National Army	0.072
Daesh	0.193	Kurdistan WP	0.282	SDF	0.036	Maydan Wardag Province	0.072
Hurriyet	0.191	Foreign Office	0.280	UN Security Council	0.034	Hamas	0.071
Ria Novosti	0.188	Us Congress	0.273	Urozgan Province	0.034	High Peace Council	0.070
National Security Council	0.183	Amnesty International	0.269	Defence Ministry	0.033	Nangarhar Province	0.067
Tass	0.183	SDF	0.266	Laghman Province	0.032	Defence Ministry	0.066
PYD	0.182	Un General Assembly	0.265	Daesh	0.030	Congress	0.063
Pakistan Army	0.179	National Security Council	0.262	Taliban Voice Of Jihad	0.029	Al Shabab	0.061
United Nations Security Council	0.174	Muslim Brotherhood	0.258	Nurestan Province	0.026	National Union Of Journalists Of Afghanistan	0.060
Un General Assembly	0.170	United Nations Security Council	0.256	Ghowr Province	0.026	Boko Haram	0.055
IRGC	0.170	Human Rights Watch	0.255	Durand Line	0.025	Senate	0.054
Al Shabab	0.165	State Duma	0.253	JuA	0.024	Shahin Army Corps	0.053
US Congress	0.165	Tass	0.248	Ministry Of Defence	0.024	Taliban Voice Of Jihad	0.052

Table 10 - Top 50 Organizations by PageRank Centrality

Unweighted				Weighted (Significance)			
Sentence		Sentence and document		Sentence		Sentence and document	
Taliban	9.5E-03	ISIL	9.2E-03	Taliban	1.9E-02	ISIL	2.1E-02
ISIL	9.5E-03	Spot Development	8.7E-03	ISIL	1.6E-02	Taliban	1.6E-02
United Nations	6.3E-03	United Nations	7.1E-03	United Nations	1.0E-02	United Nations	1.4E-02
Interior Ministry	4.1E-03	Taliban	5.9E-03	NATO	5.5E-03	Spot Development	1.3E-02
NATO	3.4E-03	Interior Ministry	4.5E-03	Kurdistan WP	4.7E-03	NATO	9.3E-03
FBI	2.9E-03	NATO	4.1E-03	Interior Ministry	4.3E-03	Interior Ministry	6.1E-03
Foreign Ministry	2.5E-03	Foreign Ministry	3.3E-03	FBI	4.2E-03	Kurdistan WP	5.1E-03
AIP	2.3E-03	FBI	2.9E-03	Hamas	4.2E-03	FBI	4.1E-03
Hamas	2.2E-03	European Union	2.7E-03	Pentagon	3.3E-03	Foreign Ministry	3.9E-03
Kurdistan WP	2.1E-03	Congress	2.4E-03	AIP	3.1E-03	Al Shabab	3.9E-03
Pentagon	2.0E-03	UN Security Council	2.3E-03	Al Shabab	2.9E-03	Pentagon	3.5E-03
Al Shabab	1.8E-03	Pentagon	2.2E-03	Voice Of Jihad	2.8E-03	Hamas	3.5E-03
Congress	1.8E-03	Senate	2.1E-03	Foreign Ministry	2.8E-03	UN Security Council	3.4E-03
Interfax Ukraine	1.8E-03	Supreme Court	2.0E-03	SDF	2.7E-03	Syrian Observatory For Human Rights	3.1E-03
Defence Ministry	1.7E-03	Al Shabab	1.8E-03	Interfax Ukraine	2.6E-03	European Union	3.0E-03
UN Security Council	1.7E-03	Defence Ministry	1.7E-03	Security Service Of Ukraine	2.5E-03	Hezbollah	2.8E-03
Boko Haram	1.7E-03	Hezbollah	1.7E-03	Hezbollah	2.3E-03	Congress	2.6E-03
Senate	1.7E-03	CIA	1.7E-03	Boko Haram	2.3E-03	SDF	2.3E-03
Hezbollah	1.6E-03	Hamas	1.6E-03	Congress	2.2E-03	Boko Haram	2.2E-03
Supreme Court	1.6E-03	State Dept.	1.6E-03	UN Security Council	2.2E-03	Security Service Of Ukraine	2.2E-03
CIA	1.5E-03	Islamic Salvation Front	1.5E-03	European Union	2.1E-03	AIP	2.2E-03
European Union	1.5E-03	PTI	1.5E-03	Senate	2.0E-03	Voice Of Jihad	2.2E-03
Express Tribune	1.3E-03	Al Qaida	1.5E-03	SBU	2.0E-03	Senate	2.2E-03
IRGC	1.3E-03	Kurdistan WP	1.5E-03	Al Qaida	2.0E-03	Supreme Court	2.1E-03
SDF	1.2E-03	Ministry Of Foreign Affairs	1.3E-03	IRGC	1.9E-03	State Dept.	2.1E-03
Security Service Of Ukraine	1.2E-03	Security Council	1.3E-03	HTS	1.9E-03	Islamic Salvation Front	2.0E-03
Ministry Of Foreign Affairs	1.2E-03	Us State Dept.	1.3E-03	CIA	1.9E-03	Interfax Ukraine	2.0E-03
State Dept.	1.2E-03	Air Force	1.2E-03	Islamic Emirate	1.9E-03	Al Qaida	2.0E-03
Air Force	1.2E-03	Reuters	1.2E-03	Supreme Court	1.8E-03	Defence Ministry	1.9E-03
Hurriyet	1.2E-03	Foreign Office	1.2E-03	Defence Ministry	1.7E-03	CIA	1.8E-03
Al Qaida	1.2E-03	AP	1.1E-03	Hurriyet	1.6E-03	Luhansk People's Republic	1.7E-03
Islamic Salvation Front	1.1E-03	Muslim Brotherhood	1.1E-03	IRGC	1.6E-03	Muslim Brotherhood	1.7E-03
Army	1.1E-03	Boko Haram	1.1E-03	Luhansk People's Republic	1.5E-03	Security Council	1.6E-03
Tass	1.1E-03	National Assembly	1.0E-03	State Dept.	1.5E-03	Hurriyet	1.6E-03
Pakistan Army	1.1E-03	AIP	1.0E-03	Pakistan Army	1.5E-03	IRGC	1.6E-03
SBU	1.0E-03	Nation	1.0E-03	Libyan National Army	1.5E-03	National Directorate Of Security	1.6E-03
Ispr	1.0E-03	Express Tribune	1.0E-03	Palestinian Authority	1.5E-03	Ministry Of Foreign Affairs	1.6E-03
PTI	1.0E-03	CCTV	9.9E-04	FSB	1.4E-03	Express Tribune	1.5E-03
Muslim Brotherhood	1.0E-03	Army	9.8E-04	Security Council	1.4E-03	PTI	1.5E-03
Ria Novosti	9.9E-04	Constitution	9.7E-04	Muslim Brotherhood	1.4E-03	IRGC	1.5E-03

Table 11 - Top 50 Persons by Degree Centrality

Unweighted				Weighted (Significance)			
Sentence		Sentence and document		Sentence		Sentence and document	
Donald Trump	2452	Donald Trump	12570	Donald Trump	11737	Donald Trump	25639
Vladimir Putin	780	Vladimir Putin	5574	Vladimir Putin	3057	Vladimir Putin	6696
Daesh	702	Barack Obama	4308	Barack Obama	1986	Bashar Al Assad	4746
Barack Obama	528	Daesh	3844	Bashar Al Assad	1472	Barack Obama	4742
Bashar Al Assad	400	Theresa May	2486	Recep T Erdogan	1312	Recep T Erdogan	3711
Muhammad	384	Bashar Al Assad	2446	Benjamin Netanyahu	1188	Daesh	3164
Ashraf Ghani	344	Ashraf Ghani	2144	James Comey	1149	Benjamin Netanyahu	2915
Benjamin Netanyahu	343	Recep T Erdogan	2123	Emmanuel Macron	1079	Ashraf Ghani	2555
Recep T Erdogan	334	Benjamin Netanyahu	2078	Rex Tillerson	1021	Emmanuel Macron	2488
Abdullah	312	Rex Tillerson	2020	Daesh	1015	Rex Tillerson	2331
Mahmoud Abbas	305	Emmanuel Macron	2015	Abdul Fattah Al Sisi	801	Abdul Fattah Al Sisi	2134
Abdul Fattah Al Sisi	281	Muhammad	2007	Ashraf Ghani	735	Haider Al Abadi	1876
Nawaz Sharif	279	Abdullah	1962	Saad Hariri	694	James Comey	1869
Emmanuel Macron	278	Angela Merkel	1555	Mahmoud Abbas	647	Theresa May	1711
Rex Tillerson	247	Nawaz Sharif	1526	Hillary Clinton	640	Bashar Assad	1640
Petro Poroshenko	235	Sergei Lavrov	1522	Michael Flynn	583	Mahmoud Abbas	1490
Hassan Rouhani	222	Haider Al Abadi	1502	Muhammad	578	Nawaz Sharif	1411
Salman	203	Bashar Assad	1434	Nawaz Sharif	537	Abu Bakr Al Baghdadi	1365
Sergei Lavrov	201	Abu Bakr Al Baghdadi	1428	Salman	537	Saad Hariri	1344
Tasnim	195	Sean Spicer	1413	Bashar Assad	530	Hassan Rouhani	1288
Theresa May	192	Hillary Clinton	1395	Sergei Lavrov	499	Hillary Clinton	1251
Muhammadammad	192	Hassan Rouhani	1261	Abdullah	486	Abdullah	1249
Bashar Assad	185	Mahmoud Abbas	1253	Jim Mattis	485	Muhammad	1245
Ramzan Kadyrov	179	Salman	1247	Hassan Rouhani	463	Sergei Lavrov	1206
Muhammad Ashraf Ghani	170	James Mattis	1203	Theresa May	450	Muhammad Ashraf Ghani	1199
James Comey	163	Petro Poroshenko	1191	James Mattis	445	Jim Mattis	1130
Haider Al Abadi	162	Boris Johnson	1168	Jared Kushner	423	James Mattis	1104
Hillary Clinton	162	Abdullah Abdullah	1121	Robertert Mueller	417	Salman Abedi	1089
James Mattis	159	George W Bush	1091	Haider Al Abadi	410	Salman	1073
Saad Hariri	154	Abdul Fattah Al Sisi	1079	Angela Merkel	389	Abdullah Abdullah	1061
Sarfraz Ahmad	151	Dmitry Peskov	1031	Salman Abedi	385	Rodrigo Duterte	1011
Babar Azam	144	Francois Hollande	992	Petro Poroshenko	368	Michael Flynn	997
Hasan Ali	144	Antonio Guterres	981	Sean Spicer	367	Petro Poroshenko	997
Gulbuddin Hekmatyar	144	Muhammadammad	960	Rodrigo Duterte	351	Angela Merkel	978
Angela Merkel	139	Narendra Modi	952	Michel Aoun	329	Sean Spicer	889
Khalifa Haftar	138	Sadiq Khan	943	Ali Abdullah Saleh	319	Ali Abdullah Saleh	877
Ghani	138	Mike Pence	928	Khalifa Haftar	315	Dmitry Peskov	832
Rodrigo Duterte	138	James Comey	902	John Mccain	314	Yeni Safak	798
Abu Bakr Al Baghdadi	136	Dmitry Medvedev	897	Devin Nunes	310	Deir Ezzor	796
Sergey Lavrov	133	John Mccain	882	Abu Sayyaf	308	Al Bab	790

Table 12 - Top 50 Persons by Eigenvector Centrality

Unweighted				Weighted (Significance)			
Sentence		Sentence and document		Sentence		Sentence and document	
Muhammadammad Ali	1	Donald Trump	1	Donald Trump	1	Donald Trump	1
Smail Ayad	0.997	Vladimir Putin	0.568	Barack Obama	0.574	Barack Obama	0.606
Najim Laachraoui	0.997	Barack Obama	0.505	Vladimir Putin	0.542	Vladimir Putin	0.534
Ibrahim El Bakraoui	0.997	Theresa May	0.381	James Comey	0.377	Bashar Al Assad	0.320
Muhammad Abrini	0.994	Rex Tillerson	0.343	Bashar Al Assad	0.238	James Comey	0.299
Khalid El Bakraoui	0.994	Sean Spicer	0.315	Benjamin Netanyahu	0.221	Recep T Erdogan	0.235
Seifeddine Rezgui	0.993	Recep T Erdogan	0.299	Recep T Erdogan	0.218	Benjamin Netanyahu	0.225
Meer Saameh Mubasheer	0.990	Bashar Al Assad	0.291	Hillary Clinton	0.197	Rex Tillerson	0.218
Rohan Imtiaz	0.990	Daesh	0.285	Rex Tillerson	0.171	Hillary Clinton	0.184
Foued Muhammad Aggad	0.990	Angela Merkel	0.279	Michael Flynn	0.157	Emmanuel Macron	0.149
Adel Kermiche	0.989	Sergei Lavrov	0.275	Emmanuel Macron	0.149	Michael Flynn	0.139
Michael Zehaf Bibeau	0.989	Emmanuel Macron	0.275	Robertert Mueller	0.113	Sean Spicer	0.127
Saleh Abdeslam	0.989	Benjamin Netanyahu	0.273	Sean Spicer	0.111	Jim Mattis	0.112
Bilal Hadfi	0.989	Boris Johnson	0.271	Jared Kushner	0.109	Bashar Assad	0.111
Rakhim Bulgarov	0.989	Hillary Clinton	0.247	Mahmoud Abbas	0.101	Theresa May	0.110
Vadim Osmanov	0.989	Muhammad	0.222	Bashar Assad	0.097	James Mattis	0.109
Ibrahim Sulayman	0.989	Malcolm Turnbull	0.221	Jim Mattis	0.093	Mahmoud Abbas	0.108
Sid Ahmad Ghlam	0.989	Francois Hollande	0.220	Theresa May	0.091	Jared Kushner	0.101
Omar Abdul Hamid El Hussein	0.989	James Mattis	0.218	Devin Nunes	0.090	Robertert Mueller	0.096
Hasanah	0.989	Abdullah	0.213	James Mattis	0.089	Dmitry Peskov	0.096
Al Jurah	0.989	Bashar Assad	0.209	Dmitry Peskov	0.082	Sergei Lavrov	0.095
Arif Sunakim	0.989	Mike Pence	0.207	Hr Mcmaster	0.080	John Mccain	0.086
Yasin Salhi	0.989	Dmitry Peskov	0.206	Steve Bannon	0.078	Hr Mcmaster	0.079
Ismail Mostefai	0.989	Salman	0.203	Sergei Lavrov	0.075	Abdul Fattah Al Sisi	0.077
Samy Amimour	0.989	George W Bush	0.201	John Mccain	0.074	Devin Nunes	0.076
Khaled Babouri	0.989	Donald J Trump	0.201	Sadiq Khan	0.072	George W Bush	0.072
Tarek Belgacem	0.989	Justin Trudeau	0.201	Abdul Fattah Al Sisi	0.067	Steve Bannon	0.071
Ines Madani	0.989	Hassan Rouhani	0.199	Salman	0.058	Sadiq Khan	0.065
Muhammad Ozturk	0.989	Sadiq Khan	0.194	Sergey Kislyak	0.055	Ashraf Ghani	0.063
Khairul Islam Paye	0.989	John Mccain	0.187	Angela Merkel	0.054	Salman	0.062
Riaz Khan Ahmadzai	0.989	Xi Jinping	0.185	Sergey Lavrov	0.051	Angela Merkel	0.062
Enes Omaragic	0.989	Mahmoud Abbas	0.183	Paul Manafort	0.050	Hassan Rouhani	0.059
Dalal Al Hashimi	0.989	Sergey Lavrov	0.182	Abdullah	0.046	Sergey Lavrov	0.058
Moussa Coulibaly	0.989	Haider Al Abadi	0.179	George W Bush	0.045	Daesh	0.057
Sarah Hervouet	0.989	Jim Mattis	0.177	Narendra Modi	0.044	Sergey Kislyak	0.056
Brahim Abdulslam	0.989	Jared Kushner	0.176	Hassan Rouhani	0.042	Mike Pence	0.056
Dian Joni Kurnaiadi	0.989	James Comey	0.175	Ashraf Ghani	0.041	Abdullah	0.052
Ahmad Muhammadazan	0.989	Michael Fallon	0.173	Mahmud Abbas	0.040	Mahmud Abbas	0.049
Ayoub	0.989	Antonio Guterres	0.168	Sayfullo Saipov	0.039	Haider Al Abadi	0.048
Safia Schmitter	0.989	Pope Francis	0.165	Mike Pence	0.036	Nikki Haley	0.046
Chakib Ahrouh	0.989	Michael Flynn	0.164	Chuck Schumer	0.035	Paul Manafort	0.045

Table 13 - Top 50 Persons by PageRank Centrality

Unweighted				Weighted (Significance)			
Sentence		Sentence and document		Sentence		Sentence and document	
Donald Trump	1.2E-02	Donald Trump	9.0E-03	Donald Trump	2.7E-02	Donald Trump	2.8E-02
Daesh	3.7E-03	Vladimir Putin	3.4E-03	Vladimir Putin	7.6E-03	Vladimir Putin	7.7E-03
Vladimir Putin	3.5E-03	Daesh	3.4E-03	Barack Obama	4.5E-03	Bashar Al Assad	5.2E-03
Barack Obama	2.2E-03	Barack Obama	2.7E-03	Daesh	3.9E-03	Barack Obama	5.0E-03
Bashar Al Assad	1.8E-03	Ashraf Ghani	1.8E-03	Bashar Al Assad	3.5E-03	Daesh	4.4E-03
Ashraf Ghani	1.6E-03	Bashar Al Assad	1.8E-03	Recep T Erdogan	3.2E-03	Recep T Erdogan	4.1E-03
Benjamin Netanyahu	1.5E-03	Theresa May	1.5E-03	Benjamin Netanyahu	2.9E-03	Ashraf Ghani	3.5E-03
Recep T Erdogan	1.4E-03	Recep T Erdogan	1.5E-03	Emmanuel Macron	2.5E-03	Benjamin Netanyahu	3.3E-03
Muhammad	1.4E-03	Abdullah	1.5E-03	Ashraf Ghani	2.4E-03	Emmanuel Macron	2.8E-03
Abdullah	1.3E-03	Muhammad	1.4E-03	Rex Tillerson	2.3E-03	Abdul Fattah Al Sisi	2.6E-03
Emmanuel Macron	1.2E-03	Benjamin Netanyahu	1.4E-03	James Comey	2.2E-03	Haider Al Abadi	2.4E-03
Mahmoud Abbas	1.2E-03	Emmanuel Macron	1.3E-03	Abdul Fattah Al Sisi	2.1E-03	Rex Tillerson	2.4E-03
Abdul Fattah Al Sisi	1.2E-03	Haider Al Abadi	1.2E-03	Nawaz Sharif	1.9E-03	Theresa May	2.0E-03
Nawaz Sharif	1.2E-03	Rex Tillerson	1.2E-03	Mahmoud Abbas	1.7E-03	Nawaz Sharif	2.0E-03
Tasnim	1.1E-03	Nawaz Sharif	1.1E-03	Muhammad	1.6E-03	Abu Bakr Al Baghdadi	1.9E-03
Petro Poroshenko	1.0E-03	Abu Bakr Al Baghdadi	1.1E-03	Saad Hariri	1.6E-03	Bashar Assad	1.9E-03
Hassan Rouhani	1.0E-03	Bashar Assad	9.8E-04	Abdullah	1.4E-03	James Comey	1.7E-03
Rex Tillerson	9.6E-04	Angela Merkel	9.1E-04	Hassan Rouhani	1.4E-03	Muhammad Ashraf Ghani	1.7E-03
Theresa May	8.5E-04	Hassan Rouhani	9.0E-04	Hillary Clinton	1.3E-03	Mahmoud Abbas	1.7E-03
Muhammad Ashraf Ghani	8.4E-04	Abdullah Abdullah	9.0E-04	Salman	1.3E-03	Abdullah	1.6E-03
Ramzan Kadyrov	8.4E-04	Mahmoud Abbas	8.8E-04	Bashar Assad	1.3E-03	Hassan Rouhani	1.6E-03
Bashar Assad	8.2E-04	Abdul Fattah Al Sisi	8.6E-04	Petro Poroshenko	1.2E-03	Muhammad	1.5E-03
Salman	8.0E-04	Salman	8.5E-04	Theresa May	1.2E-03	Abdullah Abdullah	1.4E-03
Sergei Lavrov	7.6E-04	Sergei Lavrov	8.3E-04	Sergei Lavrov	1.2E-03	Saad Hariri	1.4E-03
Gulbuddin Hekmatyar	7.4E-04	Petro Poroshenko	7.9E-04	Haider Al Abadi	1.1E-03	Petro Poroshenko	1.3E-03
Khalifa Haftar	7.4E-04	Muhammad Ashraf Ghani	7.9E-04	Salman Abedi	1.1E-03	Sergei Lavrov	1.3E-03
Rodrigo Duterte	7.2E-04	Hillary Clinton	7.6E-04	Michael Flynn	1.1E-03	Salman Abedi	1.3E-03
Abu Bakr Al Baghdadi	7.0E-04	James Mattis	7.6E-04	Jim Mattis	1.1E-03	Rodrigo Duterte	1.3E-03
Haider Al Abadi	7.0E-04	Tasnim	7.4E-04	Ramzan Kadyrov	1.1E-03	Hillary Clinton	1.2E-03
Saad Hariri	6.9E-04	Narendra Modi	7.3E-04	Khalifa Haftar	1.0E-03	James Mattis	1.2E-03
Deir Al Zour	6.8E-04	Antonio Guterres	7.3E-04	Rodrigo Duterte	1.0E-03	Salman	1.2E-03
Ghani	6.6E-04	Sean Spicer	7.2E-04	Ali Abdullah Saleh	1.0E-03	Jim Mattis	1.2E-03
Hillary Clinton	6.4E-04	Muhammadammad	7.1E-04	James Mattis	1.0E-03	Ali Abdullah Saleh	1.2E-03
Ali Abdullah Saleh	6.4E-04	Khyber Pakhtunkhwa	6.4E-04	Tasnim	1.0E-03	Angela Merkel	1.1E-03
James Comey	6.2E-04	George W Bush	6.3E-04	Angela Merkel	9.6E-04	Narendra Modi	1.1E-03
James Mattis	6.1E-04	Boris Johnson	6.3E-04	Abu Sayyaf	9.1E-04	Qamar Bajwa	1.1E-03
Muhammadammad	6.0E-04	Raqqa	6.3E-04	Muhammad Ashraf Ghani	8.9E-04	Gulbuddin Hekmatyar	1.1E-03
Angela Merkel	5.9E-04	Levant	6.2E-04	Gulbuddin Hekmatyar	8.8E-04	Hayat Tahrir Al Sham	1.0E-03
Ahmad	5.9E-04	Muhammad Morsi	6.2E-04	Narendra Modi	8.6E-04	Yeni Safak	1.0E-03
Abu Sayyaf	5.9E-04	Ghani	6.1E-04	Raila Odinga	8.6E-04	Khalifa Haftar	1.0E-03

Table 14 - Top 50 Persons and Organizations by Degree Centrality

Unweighted				Weighted (Significance)			
Sentence		Sentence and document		Sentence		Sentence and document	
Donald Trump	4252	Donald Trump	21181	Donald Trump	16299	Donald Trump	31856
Taliban	3388	ISIL	17340	Taliban	8674	ISIL	20691
ISIL	3201	Spot Development	17232	ISIL	7717	Taliban	15875
United Nations	2007	United Nations	14065	United Nations	4827	United Nations	12211
Vladimir Putin	1375	Taliban	11106	Vladimir Putin	3863	NATO	9338
Daesh	1252	Vladimir Putin	9396	NATO	3067	Spot Development	9197
Interior Ministry	1246	Interior Ministry	8676	Barack Obama	2557	Vladimir Putin	7849
NATO	990	NATO	8014	FBI	2286	Bashar Al Assad	6014
FBI	973	Barack Obama	7431	Hamas	2162	Barack Obama	5481
Barack Obama	928	Daesh	6754	Bashar Al Assad	2094	Kurdistan WP	4828
Hamas	897	Foreign Ministry	6403	Daesh	2077	Interior Ministry	4778
AIP	855	FBI	5910	Kurdistan WP	1884	Recep T Erdogan	4650
Foreign Ministry	823	European Union	5051	Recep T Erdogan	1871	Daesh	4539
Bashar Al Assad	699	Congress	4964	Hezbollah	1641	FBI	3781
Al Shabab	669	Senate	4621	Interior Ministry	1618	Hamas	3623
Recep T Erdogan	636	Pentagon	4590	Benjamin Netanyahu	1608	Benjamin Netanyahu	3526
Senate	630	UN Security Council	4564	James Comey	1588	Al Shabab	3505
Ashraf Ghani	601	Supreme Court	4233	Pentagon	1536	Syrian Observatory For Human Rights	3397
Kurdistan WP	595	Bashar Al Assad	4148	Emmanuel Macron	1506	Foreign Ministry	3193
Benjamin Netanyahu	587	Theresa May	4127	SDF	1427	Ashraf Ghani	3157
Pentagon	586	Recep T Erdogan	3702	Rex Tillerson	1364	Pentagon	3133
Congress	579	CIA	3701	Al Shabab	1340	Emmanuel Macron	3074
Hezbollah	577	Rex Tillerson	3529	Foreign Ministry	1201	Hezbollah	3035
CIA	574	Hezbollah	3490	Ashraf Ghani	1196	UN Security Council	2842
Muhammad	558	Ashraf Ghani	3489	AIP	1095	Rex Tillerson	2773
UN Security Council	537	Emmanuel Macron	3468	Congress	1079	Abdul Fattah Al Sisi	2682
Supreme Court	529	Benjamin Netanyahu	3387	Abdul Fattah Al Sisi	1055	SDF	2494
Interfax Ukraine	520	State Dept.	3359	Mahmoud Abbas	1036	Haider Al Abadi	2438
Mahmoud Abbas	498	Hamas	3344	Saad Hariri	1000	James Comey	2413
Abdul Fattah Al Sisi	498	Muhammad	3293	Senate	964	Luhansk People's Republic	2331
Nawaz Sharif	496	Abdullah	3229	Security Service Of Ukraine	934	European Union	2258
Rex Tillerson	473	Defence Ministry	3155	Voice Of Jihad	933	Congress	2145
Emmanuel Macron	470	Al Qaida	3092	UN Security Council	881	Bashar Assad	2048
Abdullah	468	AP	3030	CIA	866	Security Service Of Ukraine	2000
Defence Ministry	465	Islamic Salvation Front	2997	HTS	865	Boko Haram	1980
Islamic Salvation Front	465	Al Shabab	2976	Nawaz Sharif	865	Theresa May	1961
Boko Haram	460	Angela Merkel	2717	Boko Haram	841	Mahmoud Abbas	1947
European Union	451	Security Council	2658	Interfax Ukraine	831	Voice Of Jihad	1932
Security Service Of Ukraine	445	Nawaz Sharif	2645	Luhansk People's Republic	827	AIP	1891
IRGC	433	Reuters	2588	Bashar Assad	799	Senate	1840

Table 15 - Top 50 Persons and Organizations by Eigenvector Centrality

Unweighted				Weighted (Significance)			
Sentence		Sentence and document		Sentence		Sentence and document	
Donald Trump	1	Donald Trump	1	Donald Trump	1	Donald Trump	1
ISIL	0.730	ISIL	0.782	Barack Obama	0.508	ISIL	0.498
United Nations	0.531	United Nations	0.704	Vladimir Putin	0.462	Barack Obama	0.494
Taliban	0.460	Spot Development	0.648	ISIL	0.349	Vladimir Putin	0.439
Vladimir Putin	0.415	Vladimir Putin	0.606	James Comey	0.344	NATO	0.415
Barack Obama	0.379	Barack Obama	0.554	NATO	0.299	Taliban	0.338
Daesh	0.349	NATO	0.533	Taliban	0.271	Bashar Al Assad	0.295
NATO	0.325	Foreign Ministry	0.485	FBI	0.257	James Comey	0.254
Bashar Al Assad	0.304	Interior Ministry	0.432	Bashar Al Assad	0.224	United Nations	0.244
Foreign Ministry	0.295	Taliban	0.418	Recep T Erdogan	0.189	FBI	0.216
Recep T Erdogan	0.290	Pentagon	0.409	Benjamin Netanyahu	0.188	Recep T Erdogan	0.208
Rex Tillerson	0.285	UN Security Council	0.406	Pentagon	0.169	Pentagon	0.193
Interior Ministry	0.265	European Union	0.402	Hillary Clinton	0.168	Rex Tillerson	0.185
Pentagon	0.257	Daesh	0.394	Congress	0.167	Spot Development	0.184
UN Security Council	0.247	Congress	0.393	Rex Tillerson	0.157	Benjamin Netanyahu	0.183
Congress	0.247	Rex Tillerson	0.393	Michael Flynn	0.129	Congress	0.164
FBI	0.232	FBI	0.381	United Nations	0.125	Daesh	0.154
Hamas	0.230	Theresa May	0.374	Emmanuel Macron	0.123	Hillary Clinton	0.142
Benjamin Netanyahu	0.229	Senate	0.362	Daesh	0.110	Emmanuel Macron	0.120
CIA	0.225	State Dept.	0.354	Jim Mattis	0.102	Jim Mattis	0.113
Hezbollah	0.224	Bashar Al Assad	0.354	Voice Of Jihad	0.102	Michael Flynn	0.107
Senate	0.223	Recep T Erdogan	0.352	Robertert Mueller	0.097	Voice Of Jihad	0.107
Muhammad	0.213	CIA	0.342	Sean Spicer	0.095	Kurdistan WP	0.104
James Mattis	0.207	Angela Merkel	0.314	Jared Kushner	0.093	Bashar Assad	0.101
State Dept.	0.202	Emmanuel Macron	0.313	Bashar Assad	0.088	James Mattis	0.098
Emmanuel Macron	0.200	Hezbollah	0.312	James Mattis	0.088	Sean Spicer	0.097
Sergei Lavrov	0.198	Supreme Court	0.305	Mahmoud Abbas	0.085	State Dept.	0.090
Mahmoud Abbas	0.196	Benjamin Netanyahu	0.302	CIA	0.078	UN Security Council	0.088
Abdul Fattah Al Sisi	0.195	Sergei Lavrov	0.300	Devin Nunes	0.075	Theresa May	0.085
European Union	0.195	Security Council	0.298	Theresa May	0.073	Ashraf Ghani	0.083
Islamic Salvation Front	0.194	Sean Spicer	0.296	Justice Dept.	0.070	Mahmoud Abbas	0.083
Ashraf Ghani	0.191	Al Qaida	0.294	Senate	0.068	Haider Al Abadi	0.081
Abdullah	0.185	Us State Dept.	0.290	Sergei Lavrov	0.066	CIA	0.081
Al Qaida	0.174	Reuters	0.287	Dmitry Peskov	0.064	Hamas	0.080
Bashar Assad	0.172	AP	0.281	Hr Mcmaster	0.064	Hezbollah	0.080
Hassan Rouhani	0.165	James Mattis	0.276	Sadiq Khan	0.063	Jared Kushner	0.079
Kurdistan WP	0.164	Hillary Clinton	0.273	Steve Bannon	0.063	Robertert Mueller	0.077
Sergey Lavrov	0.164	Abdullah	0.268	Hamas	0.061	Senate	0.077
Security Council	0.163	Boris Johnson	0.268	John Mccain	0.060	Sergei Lavrov	0.075
Salman	0.163	Muhammad	0.267	Jens Stoltenberg	0.060	Jens Stoltenberg	0.073

Table 16 - Top 50 Persons and Organizations by PageRank Centrality

Unweighted				Weighted (Significance)			
Sentence		Sentence and document		Sentence		Sentence and document	
Donald Trump	6.7E-03	Spot Development	5.8E-03	Donald Trump	1.6E-02	Donald Trump	1.6E-02
Taliban	6.4E-03	Donald Trump	5.2E-03	Taliban	1.1E-02	ISIL	1.2E-02
ISIL	5.2E-03	ISIL	4.7E-03	ISIL	9.0E-03	Taliban	9.1E-03
United Nations	3.1E-03	United Nations	3.8E-03	United Nations	5.4E-03	United Nations	6.8E-03
Vladimir Putin	2.0E-03	Taliban	3.7E-03	Vladimir Putin	4.1E-03	Spot Development	6.2E-03
Interior Ministry	2.0E-03	Interior Ministry	2.4E-03	NATO	3.2E-03	NATO	5.1E-03
Daesh	2.0E-03	Vladimir Putin	2.0E-03	Barack Obama	2.5E-03	Vladimir Putin	4.2E-03
FBI	1.6E-03	NATO	1.9E-03	Daesh	2.5E-03	Interior Ministry	3.1E-03
AIP	1.5E-03	Daesh	1.9E-03	FBI	2.4E-03	Bashar Al Assad	3.0E-03
NATO	1.5E-03	Barack Obama	1.6E-03	Hamas	2.3E-03	Barack Obama	2.8E-03
Barack Obama	1.3E-03	Foreign Ministry	1.5E-03	Interior Ministry	2.3E-03	Al Shabab	2.6E-03
Hamas	1.3E-03	FBI	1.4E-03	Bashar Al Assad	2.1E-03	Daesh	2.5E-03
Al Shabab	1.3E-03	European Union	1.2E-03	Al Shabab	2.0E-03	Kurdistan WP	2.5E-03
Foreign Ministry	1.2E-03	Al Shabab	1.2E-03	Kurdistan WP	2.0E-03	Recep T Erdogan	2.3E-03
Bashar Al Assad	9.8E-04	Congress	1.1E-03	Recep T Erdogan	1.9E-03	FBI	2.0E-03
Ashraf Ghani	9.4E-04	UN Security Council	1.1E-03	Benjamin Netanyahu	1.7E-03	Hamas	1.9E-03
Boko Haram	9.1E-04	Senate	1.1E-03	Hezbollah	1.6E-03	Foreign Ministry	1.8E-03
Kurdistan WP	9.0E-04	Pentagon	1.0E-03	AIP	1.5E-03	Benjamin Netanyahu	1.8E-03
Senate	9.0E-04	Bashar Al Assad	1.0E-03	Emmanuel Macron	1.5E-03	Ashraf Ghani	1.8E-03
Recep T Erdogan	8.7E-04	Supreme Court	9.9E-04	Pentagon	1.5E-03	Syrian Observatory For Human Rights	1.6E-03
Hezbollah	8.5E-04	Ashraf Ghani	9.5E-04	Foreign Ministry	1.5E-03	Emmanuel Macron	1.6E-03
Pentagon	8.4E-04	Hamas	8.8E-04	Ashraf Ghani	1.4E-03	Abdul Fattah Al Sisi	1.5E-03
Benjamin Netanyahu	8.3E-04	Theresa May	8.7E-04	SDF	1.4E-03	Pentagon	1.5E-03
CIA	8.1E-04	Recep T Erdogan	8.6E-04	James Comey	1.4E-03	UN Security Council	1.5E-03
Congress	8.1E-04	Hezbollah	8.5E-04	Boko Haram	1.4E-03	Hezbollah	1.5E-03
Supreme Court	8.0E-04	CIA	8.2E-04	Rex Tillerson	1.3E-03	Boko Haram	1.5E-03
Interfax Ukraine	7.8E-04	Muhammad	8.1E-04	Abdul Fattah Al Sisi	1.3E-03	Haider Al Abadi	1.4E-03
UN Security Council	7.4E-04	Benjamin Netanyahu	8.1E-04	Security Service Of Ukraine	1.2E-03	Rex Tillerson	1.3E-03
Muhammad	7.4E-04	Abdullah	8.0E-04	Nawaz Sharif	1.1E-03	European Union	1.3E-03
Abdul Fattah Al Sisi	7.3E-04	Defence Ministry	8.0E-04	Congress	1.1E-03	Security Service Of Ukraine	1.2E-03
Emmanuel Macron	7.0E-04	Emmanuel Macron	7.8E-04	Interfax Ukraine	1.1E-03	AIP	1.2E-03
Defence Ministry	6.9E-04	PTI	7.6E-04	Senate	1.1E-03	SDF	1.2E-03
Nawaz Sharif	6.9E-04	Islamic Salvation Front	7.6E-04	Mahmoud Abbas	1.1E-03	Nawaz Sharif	1.1E-03
SDF	6.7E-04	State Dept.	7.3E-04	Voice Of Jihad	1.0E-03	Congress	1.1E-03
IRGC	6.6E-04	Rex Tillerson	7.3E-04	Saad Hariri	9.6E-04	James Comey	1.1E-03
Abdullah	6.6E-04	AIP	7.3E-04	UN Security Council	9.6E-04	Theresa May	1.1E-03
European Union	6.5E-04	Al Qaida	7.2E-04	Supreme Court	9.3E-04	Voice Of Jihad	1.0E-03
Islamic Salvation Front	6.5E-04	AP	7.0E-04	CIA	9.2E-04	Bashar Assad	1.0E-03
AP	6.2E-04	Haider Al Abadi	7.0E-04	HTS	8.7E-04	Senate	1.0E-03
Rex Tillerson	6.1E-04	Kurdistan WP	6.9E-04	European Union	8.7E-04	Interfax Ukraine	1.0E-03

The *Louvain* (implementation in *Networkx* [9]) and leading eigenvector community detection (implementation in *igraph* [6]) algorithms were compared.

To explain how the Louvain algorithm works, the concept of modularity must be explained. Modularity is yielded by:

$$Q = \frac{1}{2m}\sum_{i,j}\left(A_{i,j} - \frac{k_i k_j}{2m}\right)\partial(c_i, c_j)$$

where c_i denotes the community (or class) of node i and $c_i \in \{1, \dots, n_c\}$ where n_c is the total number of communities. The degree of node i is denoted k_i. The total number of edges is signified by m with the total number of edge *ends* being $2m$. Considering a scenario where edge connections are determined randomly, the chance that the one of the k_j ends is attached to vertex j is $k_j/2m$. Newman describes modularity as *a measure of the extent to which like is connected to like in a network [54]*.

The Louvain algorithm [23] works by iterating between two phases. The initial state is one where each node exists in its own community, meaning there are as many communities as nodes. The first phase considers each node i, the change in modularity, as a result of i becoming a member of the same community as each of its neighbours j, is measured. The objective is to maximize modularity. The algorithm exploits the fact that it is relatively simple to compute the change in modularity due to moving an isolated node i into a community c:

$$\Delta Q = \left[\frac{\sum_{in} + k_{i,in}}{2m} - \left(\frac{\sum_{tot} + k_i}{2m}\right)^2\right] - \left[\frac{\sum_{in}}{2m} - \left(\frac{\sum_{tot}}{2m}\right)^2 - \left(\frac{k_i}{2m}\right)^2\right]$$

where $\sum_{in}$ is the sum of the weights of the links inside c, $\sum_{tot}$ is the sum of the weights of the links incident to nodes in c, the sum of the weights of links incident node i is denoted k_i, the sum of the weights of the links from i to the nodes in c is denoted $k_{i,in}$ and the sum of all weights of links in the network is signified by m. The second phase constructs the new network whereby the nodes are now the communities found in the first phase by maximizing modularity. The weights of the links between new nodes are given by the sums of the weights of the links of the nodes in the two communities in the previous phase which are being aggregated. The Louvain algorithm is especially attractive for the purposes of this project as ΔQ is easy to compute and therefore the algorithm scales very well to large networks.

Leading eigenvector community detection also uses the modularity of the network. However, in this method the initial state is one whereby all nodes are in a single community and at each iteration each community is divide in two, if there is a possible gain in modularity, with the objective of maximizing modularity. The algorithm converges when there are no divisions that may be employed to yield a gain in modularity. The explanation has been heavily simplified so readers are directed to [54] for more clarification.

If we consider an index vector s with the same number of elements as we have nodes:

$$s_i = \begin{cases} +1 & \text{if node } i \text{ belongs to subcommunity 1} \\ -1 & \text{if node } i \text{ belongs to subcommunity 2} \end{cases}$$

then:

$$\frac{1}{2}(1 - s_i s_j) = \begin{cases} +1 & \text{if node } i \text{ and } j \text{ belong to different subcommunities} \\ -1 & \text{if node } i \text{ and } j \text{ belong to the same subcommunity} \end{cases}$$

and:

$$\partial(c_i, c_j) = \frac{1}{2}(s_i s_j + 1)$$

we can rewrite the equation for modularity as:

$$Q = \frac{1}{4m}\sum_{i,j}[A_{i,j} - P_{i,j}](s_i s_j + 1) \qquad = \frac{1}{4m}\sum_{i,j}[A_{i,j} - P_{i,j}]s_i s_j$$

and again as:

$$Q = \frac{1}{4m}\mathbf{s}^T\mathbf{B}\mathbf{s}$$

where $\mathbf{B}$ is the *modularity matrix* which is a real symmetric matrix having elements:

$$B_{i,j} = A_{i,j} - P_{i,j}$$

We rewrite $\mathbf{s}$ as a linear combination of the normalized eigenvectors $\mathbf{u}_i$ of the modularity matrix,

$\mathbf{s} = \sum_{i=1}^{n} a_i\mathbf{u}_i$ and $a_i = \mathbf{u}_i{}^T\mathbf{s}$. Then we can write:

$$Q = \frac{1}{4m}a_i^2\beta_i$$

where β_i is the eigenvalue in $\mathbf{B}$ corresponding to $\mathbf{u}_i$. At each iteration we find the most positive eigenvalue β_{max} in $\mathbf{B}$ and divide a given community into two groups depending on the signs of the corresponding vector $\mathbf{u}_{max}$. Hence:

$$s_i = \begin{cases} +1 & \text{if } u_i^{max} \geq 0 \\ -1 & \text{if } u_i^{max} < 0 \end{cases}$$

The communities detected by both algorithms were compared (Tables 17 and 18). With both algorithms the sentence and document co-occurrence rule was used. Trends were removed by applying a threshold of 0.2 standard deviations from the mean over 5 days; for the Louvain algorithm the metric used was the weighted degree and for the eigenvector community algorithm, eigenvector centrality was used. The results over four days between 02 Jun to 06 Jun 2017 can be seen on the following pages. It should also be noted that for this experiment, the entities *Donald Trump, United States* and *Spot Development* were removed as they both add a high amount of noise and are not particularly useful. In the future, a better method will be required to handle entities which occur so often in the news that they heavily affect results. For comparison, Table 16 contains terror events which occurred over the dates being examined. At this stage in the method it is only possible to match entities to GTD events, as the method needs to be extended to include phrases. However, the GTD does contain some information on entities that are related to events. Every entry in the GTD has location entities (city and country), and many have an organization (perpetrator group), but very few have persons.

These specific dates have been chosen as they cover a time window which extends either side of the date of the *London Bridge attack* [2] which occurred 03 Jun 2017. Based on earlier discussion, it was likely that this event will have been reported extensively and close to when it occurred, therefore one could expect it to exist in the corpus and, if the method has been successful so far, for it to exist as a community after the algorithms were applied. Furthermore, extending the time window either side allows one to get an idea of noise and, furthermore, how the event trends after the occurrence. It can be seen that the London Bridge attack is one of the most easily identifiable communities in both tables and its existence continues in the communities that have been detected in the days following the occurrence. Another easily identifiable event is the *Manchester Arena attack* [1] which occurred on the 22 May 2017.

It can also be seen that there are quite a few location entities in common between the terror events and the communities that have been found which occurred on the 3rd. Which could suggest that the method is successfully detecting terror events. However, when the preceding and proceeding dates are examined, it becomes apparent these dates share very few similarities. In fact, a very low threshold has been used to produce the results that are presented on purpose. The reason is that it was easy to filter out so many entities that not many more communities than the London and Manchester attacks were detected. This is an indication that there is either a very weak response from the other events or they have not made it to this stage of the method. There are a few reasons that low information related to those events may have made it this far: the SVM filtering process may have excluded documents on those events; another reason could be that the other events have not been reported by the publications we have downloaded. Another similar observation is that the community results for the 2nd Jun appear as though there has been very few documents used to generate the data for those days. This is because there are very few communities and when a stronger threshold is applied, many of the entities were removed.

Comparing the two community detection algorithms, they appear to both work appropriately well for this task. Although, it could be that there is too much noise in the data or we have not been successful in capturing the correct documents.

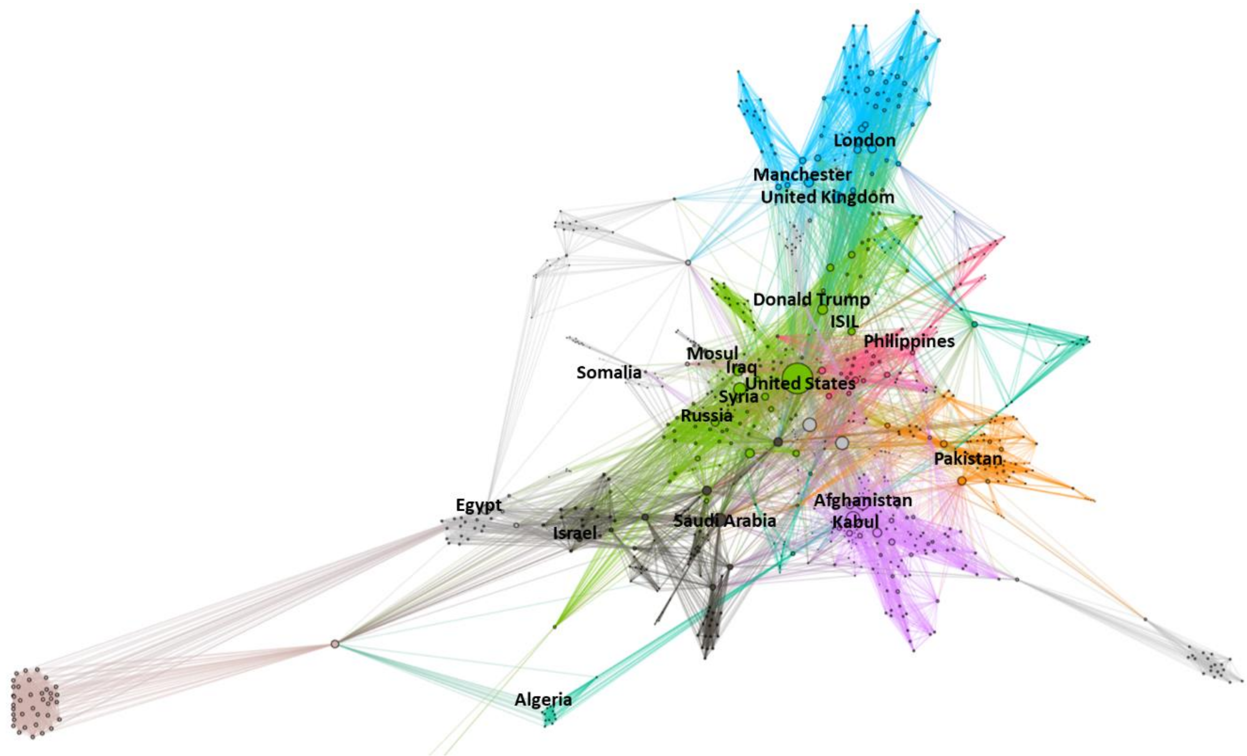


Figure 10 - Detected Communities 03 Jun 2017 by Louvain Algorithm

The above figure aims to illustrate that many of the detected entities, in common with the GTD database on the 3rd, are in different communities. It seems unlikely that this would be caused by noise so it is a reasonable indication that their occurrence is related to the events. No entities have been filtered out of the above figure. The events in Iraq and Syria are merged into a larger event whereby the most important entities are the *United States* and *Donald Trump*. It could be the case that this is a trending event of the ongoing conflicts in those regions.

Table 17 – Terror Events in GTD 02 Jun to 06 Jun 2017 – Some events which share the same country have been aggregated into a single row. This is especially apparent when there are multiple cities.

Date (Jun 2017)	Country	Cities	Perpetrator Groups
2nd	Tunisia	Sbiba district	Jund al-Khilafah
	Mali	Gao	Unknown
	Pakistan	Islamabad	Unknown
	Democratic Republic of the Congo	Mamundioma	Allied Democratic Forces (ADF)
	Bangladesh	Awami League	Awami League
	United Kingdom	Prestwich	Anti-Semitic extremists
	India	Bagabandh	Maoists
	Philippines	Marawi	Maute Group
	Iraq	Mosul	ISIL, Uknown
	Cameroon	Kolofata	Boko Haram
3rd (London bridge attack)	Iraq	Rutbah, Baaj, Mosul, Naft Khanah, Karma, Huwayrah	Badr Brigades (suspected),
	Pakistan	Rawalpindi	Khorasan Chapter of the Islamic State
	Canada	Toronto	Jihadi-inspired extremists
	Saudi Arabia	Awamiyah	Unknown
	Sri Lanka	Trincomlee	Unknown
	Mexico	Ometepec	Unknown
	Syria	Daraa	Unknown
	Philippines	Matnog district	New People's Army (NPA)
	Israel	Deir al-Assad	Unknown
	Greece	Thessaloniki	Unknown
	Nepal	Gaidatar	Communist Party of Nepal
	Algeria	Bir El Ater	Muslim extremists
	Burkina Faso	Pogwol, Petega Kourou, Peul	Ansar al-Islam (Burkina Faso)
	Somalia	Mogadishu	Al-Shabaab
	Afghanistan	Kabul	Unknown
	United Kingdom	London	Jihadi-inspired extremists
4th	Iraq	Tal Afar, Mosul, Baghdad,	ISIL, Unknown
	Sweden	Soderhamm	Uknown
	Afghanistan	Imam Sahib district, Bagram district, Kunduz, Aqbai	Taliban, Unknown
	Somalia	Mogadishu	Al-shabaab
5th	Ethiopia	Silara	PGMUD
	Pakistan	Isplinji	Unknown
	Nigeria	Awada	Unknown
	South Sudan	Loa	SPLM-IO
	Australia	Brighton	Jihadi-inspired extremists
	Somalia	Balcad, Kismayo	Al-Shabaab
	Yemen	Mukha	Houthi extremists (Ansar Allah)
	Iraq	Hadid, Balad Ruz, Tarmiyah, Baghdad, Ibrahim Ali	ISIL, Unknown
	India	Pokharibandha	(CPI-Maoist)

Table 18 – Detected Communities 02 to 06 Jun 2017

Date (Jun 2018)	Cluster size (% of Network)	Top 6 entities in cluster (Highest weighted degree)
2nd	34.8	Ukrainskaya Sluzhba Informatsii Odessa, Facebook, Council Of Civil Safety, Russian Federal Security Service, NGO
	28.8	Security Service Of Ukraine, Usi Odessa, Informatsionnyy Tsentr, Ukrainskaya Sluzhba Informatsii, Odessa Born, Serhiy Sternenko
	12.1	Sayt Goroda Odessy, Taymer, Petro Poroshenko, Dumskaya, North Atlantic Treaty Organization, Black Sea
	12.1	Bessarabia, Trassa E 95, Cabinet Of Ministers, Ukraine, Ismail, Ruslan Forostyak
	6.06	Oleksiy Chornyy, Potemkin Stairs, Istanbul, Park
	6.06	National Security And Defence Council Of Ukraine, Boeing C 135fr, Oleksandr Turchynov, Poseidon
3rd	17.0	Kabul, Afghanistan, Ashraf Ghani, United Nations, Pakistan Army, NATO
	14.1	London Ambulance Service, Theresa May, Borough Market, Bridge, West, Westminster
	11.9	Hishammuddin Hussein, Southern Philippines, Jim Mattis, Singapore Malaysia, Asia
	11.2	United Kingdom, Ariana Grande, Manchester, Salman Abedi, Manchester Arena, London
	9.0	Kseniya Kirillova, Russia, Washington, Alexander Shchetinin, Ukraine, Riyad Haddad
	8.7	Iran, Muhammad Reza Tabesh, US Central Intelligence Agency, Paris, Narendra Modi, Turkey
	8.3	Saudi Arabia, Israel, Arab, Fawaz Gerges, London School Of Economics, Egypt
	8.3	Pakistan Institute Of Peace Studies, Syed Ali Shah Geelani, National Investigation Agency, Ishan Ghani, India, Kashmir
	6.4	Syria, United Nations, Iraq, ISIL, Raqqa, Syrian Observatory For Human Rights
4th	20.8	London, Bridge, Borough Market, Sadiq Khan, Westminster, Gerard Vowls
	17.1	Vladimir Putin, Russia, St Petersburg, Europe, Dmitry Peskov, Kremlin,
	9.9	Manchester, Ariana Grande, England, European Union, Salman Abedi, Manchester Arena
	9.6	United Kingdom, Michel Aoun, Great Britain, Foreign Ministry, State Of The Union, Middle East
	8.3	Pakistan, India, Sarfraz Ahmad, Virat Kohli, Shadab Khan, Hardik Pandya
	7.8	Syria, ISIL, Iraq, West, State Of The Union, Henry Jackson Society
	6.0	France, Emmanuel Macron, Jean Yves Le Drian, Narendra Modi, France Info Radio, Elysee Palace
	5.0	Real Madrid, Bangladesh, New Zealand, Steve Smith, Australia, Manchester United
	4.2	Jeremy Corbyn, Labour, Conservatives , Labour And Liberal Democrats, Nicola Sturgeon, Conservatives , Labour, Opposition Labour Party
	3.5	Foreign Ministry, Ministry Of Roads And Urban Development, Ministry Of The Youth And Sports, Italy, Iran, Bahram Qassemi
5th	22.3	Khuram Shazad Butt, United Kingdom, Mark Rowley, Al Muhammadajiroun, Pakistan, Jihadis Next Door
	17.8	East London, Bridge, Britain, Borough Market, Barking, Foreign Ministry
	12.4	Rachid Redouane, Khuram Butt, Ireland, Dublin, Rachid Elkhdar, Scotland
	11.4	Jean Yves Le Drian, St Thomas' Hospital, France, New Zealand, Australia, Andrew Morrison
	10.8	ISIL, British Transport Police, Florin Morariu, Bread Ahead, Westminister Bridge, Borough High Street And Market
	8.3	Manchester, Cressida Dick, Mayor Sadiq Khan, Westminster, Ariana Grande, Jeremy Corbyn
	6.7	Manchester Arena, United Arab Emirates, Grande, One Love Manchester, Jibril Palomba, Cold Play
	6.0	Theresa May, Vladimir Putin, John Kerry, Google, Brexit, NBC
	3.8	Christine Archibald, Europe, Saint Petersburg, British Columbia, Cassie Ferguson Rowe, Berlin

Table 19 - Communities Detected by Eigenvector Algorithm

Date (Jun 2018)	Cluster size (% of Network)	Top 6 entities in cluster (Highest weighted eigen centrality)
2nd	48.5	Ukrainskaya Sluzhba Informatsii Odessa, Facebook, Security Service Of Ukraine, Taymer, Petro Poroshenko, Bessarabia, Dumskaya
	21.2	NATO, Black Sea, Romania, Thessaloniki, Greece, Poland
	19.0	Germany, Interior Ministry Postpones Collective Deportations, Christian Democratic Union, Federal Interior Minister De Maiziere
	11.4	Atlantic Resolve, Eucom Commander Army, European Reassurance Initiative, Marine Corps, Air Force, Curtis M Scaparrotti, David Allvin
3rd	19.2	Pakistan Institute Of Peace Studies, Ishan Ghani, Muhammadammad Amir Rana, Paris, National Counter Terrorism Authority, Nacta
	14.4	Syria, Raqqa, Iraq, Saudi Arabia, Syrian Observatory For Human Rights, SDF
	13.1	Muhammad Reza Tabesh, Hishammuddin Hussein, Asia, Irna, Southern Philippines, Malaysia
	12.8	Ashraf Ghani, Afghanistan, NATO, Pashto, Muhammad Salim Ezadyar, Abdullah Abdullah, Taliban
	10.6	London Ambulance Service, Bridge, West, Westminster, Theresa May, Manchester, Mark Rowley, Bridge Tube, Borough Market
	9.6	Russia, Kseniya Kirillova, Washington, Ukraine, Alexander Shchetinin, Riyad Haddad, Moscow, Alya Shandra, Ria Novosti, Mikhail Bogdanov
	5.8	Iran, Alavi Foundation, Manhattan, John Gleeson, New York Civil Liberties Union, Sayyed Musawi, Katherine Forrest, Model United Nations
	5.8	Kabul, Narendra Modi, Hassan Rouhani, Guilds, Fazel, Us Central Intelligence Agency, Tehran, Ayatollah Ruhollah Khomeini
	5.8	Gul Nabi Ahmadzai, United Nations, Asif Ashna, Salim Ezadyar, Waheed Majroh, Najib Danish, Interior Ministry
	2.9	Pakistan Army, Haqqani, Qamar Javed Bajwa, United Arab Emirates, Aizaz, Ahmad Chaudhry, Post, Kandahar, Helmand
4th	29.6	London, Bridge, Borough Market, Britain, Sadiq Khan, Theresa May, Westminster, Jeremy Corbyn, Justin Trudeau, Downing Street
	24.3	Vladimir Putin, Syria, ISIL, Iraq, Dmitry Peskov, Kremlin, Foreign Ministry, West, Federation Council Committee On Defence And Security
	12.5	Manchester, Ariana Grande, England, Real Madrid, Manchester United, Michael Carrick, New Zealand, Manchester Arena
	9.19	Emmanuel Macron, France, Paris, Berlin, Europe, Narendra Modi, Nice, Russia, Christophe Castaner, Saint Petersburg
	8.3	Pakistan, Nawaz Sharif, Muhammadammad Shahbaz Sharif, APP, India, Edgbaston, Pm House, Lahore, Punjab, Sarfraz Ahmad
	5.6	United Kingdom, Great Britain, Kuwait, Novak Djokovic, Ministry Of Foreign Affairs, Nicola Sturgeon, Iraqi Ministry Of Foreign Affairs
	2.9	Duncan Smith, Liberal Democrats, Federation, Police, Lord Carlile, Myriam Francois, Tpims, Scottish National, Scottish Labour, Berriew
	1.7	Alison Mutler, Jo Kearney, Raphael Satter, David Keaton, Florin Morariu, Bucharest, Bread Ahead, Barclays, Oi
	0.9	Michel Aoun, Queen Elizabeth Ii, Lebanon, NNA, Beirut
	0.7	Salman Bin Abdulaziz, Kingdom Of Saudi Arabia, SPA, Riyadh
5th	22.3	Rachid Redouane, Khuram Shazad Butt, Mark Rowley, Pakistan, Khuram Butt, ISIL, United Kingdom, Al Muhammadajiroun, Ireland
	17.8	East London, Bridge, Borough Market, Britain, Barking, Manchester, Cressida Dick, Mayor Sadiq Khan, Westminster, France
	12.4	Christine Archibald, St Thomas' Hospital, Spain, Ferguson, Tyler, City Hall, Cassie Ferguson Rowe, Candice Hedge, British Columbia
	11.4	Theresa May, Australia, Europe, England, Jean Yves Le Drian, Ariana Grande, Jeremy Corbyn, New Zealand, Pharrell Williams, Justin Bieber
	10.8	Westminister Bridge, Florin Morariu, Bread Ahead, Associated, Lori Hinnant, Raphael Satter, Alison Mutler, Niko Price, Sylvia Hui
	8.3	Germany, Borough High Street And Market, Elizabeth Fry, Southern And Thameslink, Montague Close, Stoney Street, National Rail
	6.7	Myriam Francois Mehri Niknam, Joseph Interfaith Foundation, Muhammadammad Habibur Rahman, Mak Chishty, Justin Welby, Canterbury

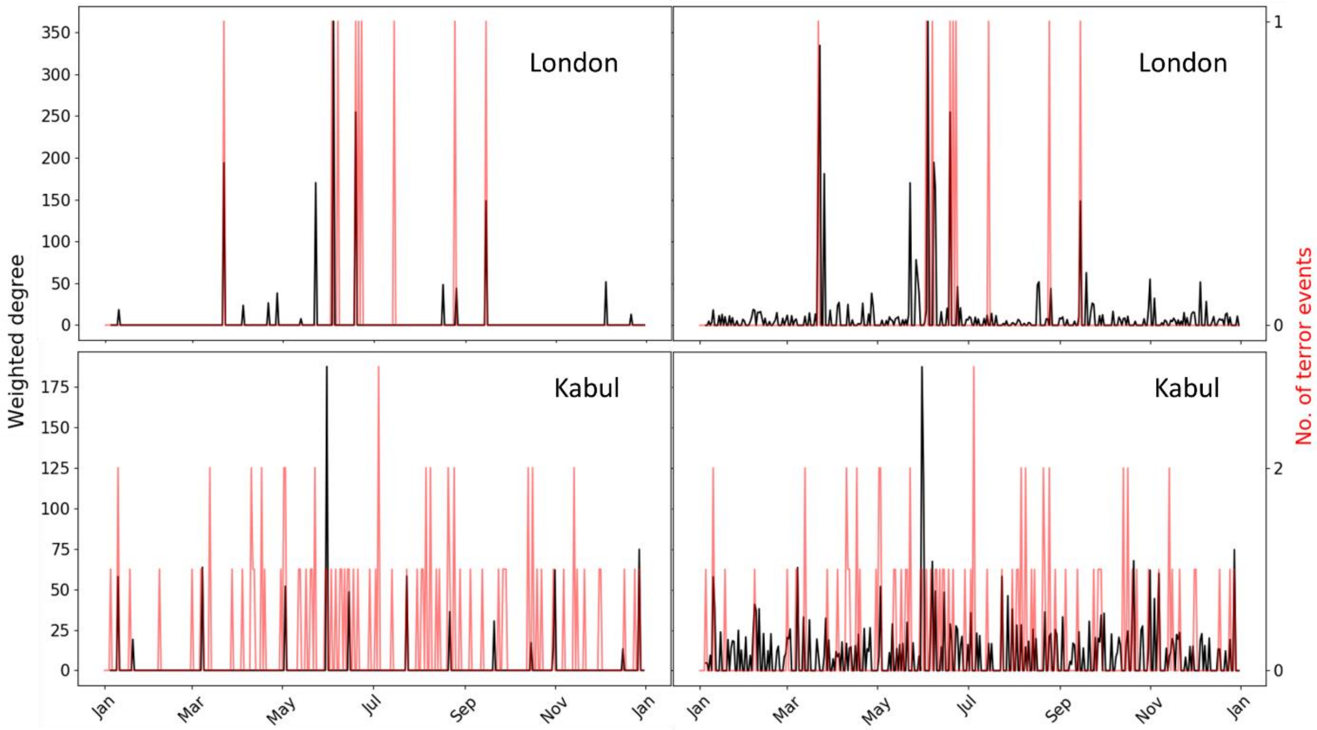


Figure 11 - Weighted degree versus number of terror events

Top: London. Bottom: Kabul. Thresholds of 1.9 and 0.5 standard deviations have been applied *left* and *right* respectively. Window size: 7 days. Co-occurrence rule: sentence and document level.

The above figure highlights one of the disadvantages of thresholding all entities by the same threshold. It can be seen that terror events seldom occur in London when compared to Kabul, where they are approaching a daily occurrence. The observed response that occurs when a terror event occurs in Kabul is not very high in comparison to its average state. Conversely, the response that occurs when a terror event occurs in London is much higher. In fact, the normal 'resting' state of both entities is roughly equivalent. The few strong responses seen for London often line up with the occurrence of a terror event, which is a good indication that useful information about events occurring in London has been captured. But it is difficult to say the same for Kabul. When an appropriate threshold that fits well for London is applied e.g. 1.9 std., a lot of the response for Kabul is lost, whereas if a weaker threshold is applied, as to allow some response for Kabul into the network, more noise from the London entity is detected. This disparity could be due to imbalance reporting or an indication that the filtering method treats terror events in both countries unequally. It seems that there is a reasonable response for London but a low response for Kabul, which could be being masked by noise.

5 Discussion and Conclusion

The development and subsequent analysis of results of a potential TSB detection mode system specially designed for detecting terror events has been outlined. Although, the project has not gone as far as being able to fully detect events. The process has been explored and many things have been learned.

The usage of Doc2Vec was probably an oversimplification because it is possible that a document can contain multiple topics. For example, sometimes a document may be a recount of events that have occurred over the day, which may include many subjects including terrorism. The problem with this is that, in the vector space into which the Doc2Vec model projects, although the terrorism topic on its own may exists at a point within the positive region of the SVM's decision boundary, the other topics may not. The *document's* vector, as a result of the non-terror related topics, may exist on the negative side on the decision boundary. This could also work the opposite way. Doc2Vec may be better suited to this type of use when using a medium such as Tweets, which are on average much shorter documents and likely contain fewer topics per document. The reason that the SVM filter worked as well as it did may be because terror related documents may rarely contain multiple topics. To work towards a better method for this there are a few approaches one could take. The documents could be broken up into paragraphs. Humans tend to separate sections of a document that discuss different things into different paragraphs. Then only paragraphs which are classified as important may be extracted. However, this method could miss important context it one is aiming to create any type of summarization of an event. Another approach could be to classify all of the paragraphs in a document, and if a single paragraph is classified as terror related, the whole document is classified as terror related. This would increase the sensitivity of the detector but also increase noise somewhat.

The higher tendency of the Random Forest and Ada boost classifiers to misclassify the positive class during cross validation may be attributable to the imbalance in the training data sets. The ratio of negative to positive samples was approximately 3:1. There are a few ways in which the imbalance could be alleviated somewhat. The best way is to collect more positive samples as the only change is that we gain more information on the positive class. It is possible to *undersample* by removing some random negative samples to make the dataset more equal, but this results in some loss of information on the negative class and is usually not as good as *oversampling*. Oversampling is when we create synthetic instances of the positive class to balance the dataset. This can often give better performance than undersampling as, mentioned earlier, there is no loss of information from the dataset. The most popular technique for this is *Synthetic Minority Oversampling Technique* (SMOTE) [28] which uses a modified version of the well-known *k-Nearest Neighbours* clustering algorithm. SMOTE basically attempts to interpolate rather than extrapolate the positions of new samples by filling in the 'between' regions of the class to be oversampled. What is meant by this is that the regions that lie between the current instances of the class to be oversampled, which have a higher probability of being regions where real samples may lie if they were included in the dataset. Another consideration is that the ROC performance metric, which was used to evaluate performance of the classifiers, is not the best metric for working with unbalanced data. In hindsight, it may have been more appropriate to use the AUC of the Precisions-Recall curve. Precision is the ratio of the number of true positives divided by the sum of the true positives and false positives (4). It describes how good a model is at predicting the positive class. Recall is calculated as the ratio of the number of true positives divided by the sum of the true positives and the false negatives (5). Using both precision and recall is useful in cases where there is an imbalance in the observations between the two classes. Specifically, if there are many examples of no event (class 0) and only a few examples of an event (class 1). Using the AUC of this metric may have been better than using AUC of the ROC curve.

$$precision = \frac{true\ positives}{true\ positives\ +\ false\ negatives} \tag{4}$$

$$recall = \frac{true\ positives}{true\ positives +\ false\ negatives} \tag{5}$$

It was apparent that when reviewing the GTD data that a vast majority of terror events occur in countries that do not have English as their predominant language. Taking this into consideration, there may be advantages to using other languages in the pursuit of detecting terror events because a large portion of news media that reports on terror events may not be in English. This is something worth exploring in future work. Though, it may not be the case that there is a single language that is used as much as English between the different countries in these regions.

So far only sentence and document co-occurrence rules have been investigated. It could be an advantage to also experiment with paragraphs. As mentioned earlier, humans tend to compartmentalize sections with different meaning into paragraphs.

When looking at community structure, it was apparent that there was either a low response for some events, they had not been captured adequately by data collection i.e. were not reported by the publications which were used. A simple way to remedy this low performance could be to use more documents at the start. This could be done by adding more publications to the corpus. Theoretically, as long as the SVM step is working as intended, adding more documents should increase the response and at the same time reduce relative noise. However, if the problem is that the SVM is not classifying documents about those events then the classifier will have to be retrained. To ensure that the SVM is not the issue, some documents for the undetected events will need to be found and how the SVM classifies them observed.

A fundamental problem with the Louvain and eigenvector community detection algorithms is that they do not allow an entity to exist in more than one community. This is different to real-life whereby an entity can exists in multiple communities. For example, a perpetrator group may be connected to multiple terror events which occur simultaneously. Another example, that has been observed in the GTD, is that more than one terror event can occur in a country simultaneously. However, in each the entities are being forced into a single community, possibly making it more difficult to detect the other.

One way to solve this problem is to increase the level of precision by considering smaller time windows. In fact, one way to interpret the problem is that it is caused by the consideration of time windows which are too large. So far, only networks of entire days have been created. Terror events which have common entities, most likely, do not occur exactly simultaneously, and may not be reported simultaneously, but only appear to do so as we cannot distinguish between time intervals smaller than a day. However, as we have seen with the community detection experiments and figure 7, there were few documents for some days. It would be necessary to acquire a larger corpus to consider smaller time intervals. It would also be beneficial to use online documents as it was observed, these appear to always have a timestamp, which is necessary to consider time intervals smaller than a day. It would be appropriate for any future implementation to incorporate these improvements.

Another approach to solve the common entities problem is to use an adapted community detection algorithm that outputs probabilities for each entity belonging to each community. This could also be used with centrality metrics to quantify the importance of the entity to each community. A threshold could be used to determine if an entity belong to each community. This would allow an entity to belong to multiple communities.

So far only entities have been used to build networks but in the future it would be advantageous to extend the networks with frequent phrases as we have seen in previous work [49; 53]. This could also be a way to alleviate the common entity problem discussed above.

6 References

[1] BBC. 2017. Manchester Arena attack: What happened?. [Accessed 03 Aug 2019]. URL= https://www.bbc.co.uk/newsround/40009766.

[2] BBC. 2019. London Bridge attack: What happened. [Accessed 03 Aug 2019]. URL= https://www.bbc.co.uk/news/uk-england-london-40147164.

[3] Dateutil. [Accessed 08 Aug 2019]. URL= https://dateutil.readthedocs.io/en/stable/.

[4] Doc2Vec. 2016. [Accessed 16 Jun 2019]. URL= https://github.com/jhlau/doc2vec.

[5] eMarketer. 2019. Number of Social Media Users Worldwide from 2010 to 2021 (in billions). [Online, Accessed: 15 Jul 2019]. URL= https://www.statista.com/statistics/278414/number-of-worldwide-social-network-users/.

[6] igraph. Network analysis tool. [Accessed 30 Aug 2019]. URL= https://igraph.org/python/.

[7] National Consortium for the Study of Terrorism and Responses to Terrorism (START). (2018). Global Terrorism Database [Data file]. URL= https://www.start.umd.edu/gtd.

[8] Natural Language Toolkit. URL= http://www.nltk.org/.

[9] Networkx. Network analysis tool. [Accessed 03 Aug 2019]. URL= https://networkx.github.io/.

[10] OntoNotes. 2013. [Accessed 05 Aug 2019]. URL= https://catalog.ldc.upenn.edu/LDC2013T19.

[11] pytz. [Accessed 08 Aug 2019]. URL= http://pytz.sourceforge.net/.

[12] Reporting on Suicide. 2017. [Accessed 04 Aug 2019]. URL= http://reportingonsuicide.org/.

[13] Scikit-Learn. [Accessed 08 Aug 2019]. https://scikit-learn.org/stable/.

[14] Selenium. [Accessed 08 Aug 2019]. URL= https://selenium-python.readthedocs.io/.

[15] Simplemaps. [Accessed 20 Jul 2019]. URL= https://simplemaps.com/data/world-cities.

[16] Theune, C. pycountry. 2019. [Accessed 15 Jul 2019]. URL= https://pypi.org/project/pycountry/.

[17] Aggarwal, C. and Subbian, K., 2012. Event Detection in Social Streams. In *Proceedings of the 2012 SIAM International Conference on Data Mining* Society for Industrial and Applied Mathematics, 624-635. DOI= http://dx.doi.org/10.1137/1.9781611972825.54.

[18] Alex, K., Sutskever, I., and Hinton, G.E., 2012. ImageNet Classification with Deep Convolutional Neural Networks, 1097--1105.

[19] Allan, J., Carbonell, J., Doddington, G., Yamron, J., and Yang, Y., 2000. Topic Detection and Tracking Pilot Study Final Report(11/13).

[20] Becker, H., Naaman, M., and Gravano, L., 2011. Beyond trending topics: Real-world event identification on twitter. *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM11)*, 1-17.

[21] Beckmann, K., Dewenter, R., and Thomas, T., 2017. Can News Draw Blood? The Impact of Media Coverage on the Number and Severity of Terror Attacks? *23*, 1, 1-16. DOI= http://dx.doi.org/10.1515/peps-2016-0025.

[22] Bik, H.M. and Goldstein, M.C., 2013. An Introduction to Social Media for Scientists. *PLoS biology 11*, 4, e1001535-e1001535. DOI= http://dx.doi.org/10.1371/journal.pbio.1001535.

[23] Blondel, V., Guillame, J., Lambiotte, R., and Lefebvre, E., 2008. Fast Unfolding of Communities in Large Networks. *Journal of Statistical Mechanics: Theory and Experiment*. DOI= http://dx.doi.org/doi:10.1088/1742-5468/2008/10/P10008.

[24] Bonacich, P. and Lloyd, P., 2001. Eigenvector-like measures of centrality for asymmetric relations. *Social Networks 23*, 3 (2001/07/01/), 191-201. DOI= http://dx.doi.org/https://doi.org/10.1016/S0378-8733(01)00038-7.

[25] Boser, B., Guyon, I., and N. Vapnik, V., 1996. A Training Algorithm for Optimal Margin Classifier. *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory 5*(08/09). DOI= http://dx.doi.org/10.1145/130385.130401.

[26] Bradley, A.P., 1997. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recogn. 30*, 7, 1145-1159. DOI= http://dx.doi.org/10.1016/s0031-3203(96)00142-2.

[27] Breiman, L., 2001. Random Forests. *Machine Learning 45*, 1 (2001/10/01), 5-32. DOI= http://dx.doi.org/10.1023/A:1010933404324.

[28] Chawla, N., Bowyer, K., O. Hall, L., and Philip Kegelmeyer, W., 2002. SMOTE: Synthetic Minority Over-sampling Technique. *J. Artif. Intell. Res. (JAIR) 16*(01/01), 321-357. DOI= http://dx.doi.org/10.1613/jair.953.

[29] Cui, W., Wang, P., Du, Y., Chen, X., Guo, D., Li, J., and Zhou, Y., 2017. An algorithm for event detection based on social media data. *Neurocomputing 254*(2017/09/06/), 53-58. DOI= http://dx.doi.org/10.1016/j.neucom.2016.09.127.

[30] Dode, A. and Hasani, S., 2017. PageRank Algorithm. *10.9790/0661-1901030107 19*(02/09), 2278-2661. DOI= http://dx.doi.org/10.9790/0661-1901030107.

[31] E. Schapire, R., 2013. Explaining AdaBoost, 37-52. DOI= http://dx.doi.org/10.1007/978-3-642-41136-6_5.

[32] Enders, W., Sandler, T., and Gaibulloev, K., 2011. Domestic Versus Transnational Terrorism: Data, Decomposition, and Dynamics. *Journal of Peace Research 48*(05/01), 319-337. DOI= http://dx.doi.org/10.1177/0022343311398926.

[33] Garg, M., 2016. Review on Event Detection Techniques in Social Multimedia. *Online Information Review 40*, 3, 347-361. DOI= http://dx.doi.org/10.1108/OIR-08-2015-0281.

[34] Gomide, J., Veloso, A., Meira Jr, M., Almeida, V., Benevenuto, F., Ferraz, F., and Teixeira, M., 2011. Dengue surveillance based on a computational model of spatio-temporal locality of Twitter. In *Proceedings of the Proceedings of the 3rd International Web Science Conference* (Koblenz, Germany2011), ACM, 1-8. DOI= http://dx.doi.org/10.1145/2527031.2527049.

[35] Gordon, T., Sharan, Y., and Florescu, E., 2015. Prospects for Lone Wolf and SIMAD terrorism. *Technological Forecasting and Social Change 95*(2015/06/01/), 234-251. DOI= http://dx.doi.org/10.1016/j.techfore.2015.01.013.

[36] Grolmusz, V., 2015. A note on the PageRank of undirected graphs. *Information Processing Letters 115*, 6 (2015/06/01/), 633-634. DOI= http://dx.doi.org/https://doi.org/10.1016/j.ipl.2015.02.015.

[37] Gruhl, D., Guha, R., Liben-Nowell, D., and Tomkins, A., 2004. Information Diffusion Through Blogspace. *In Proceedings of the 13th International World Wide Web Conference*, 491–501.

[38] Guohui, L., Song, L., Xudong, C., Hui, Y., and Heping, Z., 2014. Study on Correlation Factors that Influence Terrorist Attack Fatalities Using Global Terrorism Database. *Procedia Engineering 84*(2014/01/01/), 698-707. DOI= http://dx.doi.org/https://doi.org/10.1016/j.proeng.2014.10.475.

[39] Hachey, B., Radford, W., Nothman, J., Honnibal, M., and Curran, J.R., 2013. Evaluating Entity Linking with Wikipedia. *Artificial Intelligence 194*(2013/01/01/), 130-150. DOI= http://dx.doi.org/10.1016/j.artint.2012.04.005.

[40] Han Lau, J. and Baldwin, T., 2016. *An Empirical Evaluation of doc2vec with Practical Insights into Document Embedding Generation*.

[41] Hewage, T.N., Halgamuge, M., Syed, A., and Ekici, G., 2018. *Review: Big data techniques of google, Amazon, Facebook and Twitter*.

[42] Jenkins, B.M., Willis, H.H., and Han, B., 2016. Do Significant Terrorist Attacks Increase the Risk of Further Attacks? *Initial Observations from a Statistical Analysis of Terrorist Attacks in the United States and Europe from 1970 to 201*. DOI= http://dx.doi.org/https://doi.org/10.7249/PE173.

[43] Jetter, M., 2017. Terrorism and the Media: The Effect of US Television Coverage on Al-Qaeda Attacks. *I Z A Institute of Labour Economics 10708*.

[44] Jolliffe, I.T. and Cadima, J., 2016. Principal component analysis: a review and recent developments. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences 374*, 2065, 20150202-20150202. DOI= http://dx.doi.org/10.1098/rsta.2015.0202.

[45] Kearns, E.M., Betus, A.E., and Lemieux, A.F., 2019. Why Do Some Terrorist Attacks Receive More Media Attention Than Others? *Justice Quarterly*, 1-24. DOI= http://dx.doi.org/10.1080/07418825.2018.1524507.

[46] Lafferty, J., Mccallum, A., and Pereira, F., 2001. *Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data*.

[47] Le, Q. and Mikolov, T., 2014. Distributed representations of sentences and documents. In *Proceedings of the Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32* (Beijing, China2014), JMLR.org, 3045025, II-1188-II-1196.

[48] Mcminn, A.J., Moshfeghi, Y., and Jose, J.M., 2013. Building a large-scale corpus for evaluating event detection on twitter. In *Proceedings of the Proceedings of the 22nd ACM international conference on Information & Knowledge Management* (San Francisco, California, USA2013), ACM, 2505695, 409-418. DOI= http://dx.doi.org/10.1145/2505515.2505695.

[49] Melvin, S., Yu, W., Ju, P., Young, S., and Wang, W., 2017. Event Detection and Summarization Using Phrase Network, 89-101. DOI= http://dx.doi.org/10.1007/978-3-319-71273-4_8.

[50] Mickolus, E.F., Sandler, T., Flemming, P.A., and Simmons, S.L., 2016. The International Terrorism: Attributes of Terrorist Events (ITERATE) dataset, I. VINYARD SOFTWARE Ed. Scholars Portal Dataverse. DOI= http://dx.doi.org/doi:10.5683/SP/YRFU12.

[51] Mikolov, T., Chen, K., Corrado, G.S., and Dean, J., 2013. Efficient Estimation of Word Representations in Vector Space. *CoRR abs/1301.3781*.

[52] Moutidis, I. and Williams, H.T.P., 2019. Named entity driven event detection for social and conventional news streams. (Unpublished manuscript).

[53] Moutidis, I. and Williams, H.T.P., 2019. Towards a complex network approach to detecting events in high-volume news streams. (Unpublished manuscript).

[54] Newman, M.E.J., 2006. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E 74*, 3 (09/11/), 036104. DOI= http://dx.doi.org/10.1103/PhysRevE.74.036104.

[55] Newman, M.E.J., 2010. *Networks: An Introduction*. Oxford University Press.

[56] Panagiotou, N., Katakis, I., and Gunopulos, D., 2016. Detecting Events in Online Social Networks: Definitions, Trends and Challenges. In *Solving Large Scale Learning Tasks. Challenges and Algorithms: Essays Dedicated to Katharina Morik on the Occasion of Her 60th Birthday*, S. MICHAELIS, N. PIATKOWSKI and M. STOLPE Eds. Springer International Publishing, Cham, 42-84. DOI= http://dx.doi.org/10.1007/978-3-319-41706-6_2.

[57] Platt, J., 2000. Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods. *Adv. Large Margin Classif. 10*(06/23).

[58] Ray, P.P., 2018. A survey on Internet of Things architectures. *Journal of King Saud University - Computer and Information Sciences 30*, 3 (2018/07/01/), 291-319. DOI= http://dx.doi.org/10.1016/j.jksuci.2016.10.003.

[59] Reul, C., Springmann, U., Wick, C., and Puppe, F., 2018. *State of the Art Optical Character Recognition of 19th Century Fraktur Scripts using Open Source Engines*.

[60] Sakaki, T., Okazaki, M., and Matsuo, Y., 2010. Earthquake shakes Twitter users: real-time event detection by social sensors. In *Proceedings of the Proceedings of the 19th international conference on World wide web* (Raleigh, North Carolina, USA2010), ACM, 1772777, 851-860. DOI= http://dx.doi.org/10.1145/1772690.1772777.

[61] Schinas, M., Papadopoulos, S., Kompatsiaris, Y., and Mitkas, P.A., 2018. Event Detection and Retrieval on Social Media. *CoRR abs/1807.03675*(/).

[62] Sutton, C. and Mccallum, A., 2012. An Introduction to Conditional Random Fields. *Found. Trends Mach. Learn. 4*, 4, 267-373. DOI= http://dx.doi.org/10.1561/2200000013.

[63] Trovati, M., 2018. Mining Social Media: Architecture, Tools and Approaches to Detecting Criminal Activity. In *Applications of Big Data for National Security*, 155-170.

[64] Van Der Maaten, L.J.P. and Hinton, G.E., 20019. Visualizing High-Dimensional Data Using t-SNE. *Journal of Machine Learning Research 9*, 2579-2605.

[65] Yardi, S. and Boyd, D., 2010. Tweeting from the Town Square: Measuring Geographic Local Networks. *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media*, 194-201.