Event detection in finance using hierarchical clustering algorithms on news and tweets (#55204)

First revision

Guidance from your Editor

Please submit by **17 Feb 2021** for the benefit of the authors .

Structure and Criteria
Please read the 'Structure and Criteria' page for general guidance.

Raw data check
Review the raw data.

Image check
Check that figures and images have not been inappropriately manipulated.

Privacy reminder: If uploading an annotated PDF, remove identifiable information to remain anonymous.

Files

Download and review all files from the materials page.

1 Tracked changes manuscript(s)
1 Rebuttal letter(s)
16 Figure file(s)
1 Latex file(s)

Structure and Criteria

Structure your review

The review form is divided into 5 sections. Please consider these when composing your review:

1. **BASIC REPORTING**
2. **EXPERIMENTAL DESIGN**
3. **VALIDITY OF THE FINDINGS**
4. General comments
5. Confidential notes to the editor

You can also annotate this PDF and upload it as part of your review

When ready submit online.

Editorial Criteria

Use these criteria points to structure your review. The full detailed editorial criteria is on your guidance page.

BASIC REPORTING

- Clear, unambiguous, professional English language used throughout.
- Intro & background to show context. Literature well referenced & relevant.
- Structure conforms to PeerJ standards, discipline norm, or improved for clarity.
- Figures are relevant, high quality, well labelled & described.
- Raw data supplied (see PeerJ policy).

EXPERIMENTAL DESIGN

- Original primary research within Scope of the journal.
- Research question well defined, relevant & meaningful. It is stated how the research fills an identified knowledge gap.
- Rigorous investigation performed to a high technical & ethical standard.
- Methods described with sufficient detail & information to replicate.

VALIDITY OF THE FINDINGS

- Impact and novelty not assessed. Negative/inconclusive results accepted. *Meaningful* replication encouraged where rationale & benefit to literature is clearly stated.
- All underlying data have been provided; they are robust, statistically sound, & controlled.
- Speculation is welcome, but should be identified as such.
- Conclusions are well stated, linked to original research question & limited to supporting results.

Standout reviewing tips

The best reviewers use these techniques

Tip	*Example*
Support criticisms with evidence from the text or from other sources	*Smith et al (J of Methodology, 2005, V3, pp 123) have shown that the analysis you use in Lines 241-250 is not the most appropriate for this situation. Please explain why you used this method.*
Give specific suggestions on how to improve the manuscript	*Your introduction needs more detail. I suggest that you improve the description at lines 57- 86 to provide more justification for your study (specifically, you should expand upon the knowledge gap being filled).*
Comment on language and grammar issues	*The English language should be improved to ensure that an international audience can clearly understand your text. Some examples where the language could be improved include lines 23, 77, 121, 128 – the current phrasing makes comprehension difficult.*
Organize by importance of the issues, and number your points	*1. Your most important issue* *2. The next most important item* *3. …* *4. The least important points*
Please provide constructive criticism, and avoid personal opinions	*I thank you for providing the raw data, however your supplemental files need more descriptive metadata identifiers to be useful to future readers. Although your results are compelling, the data analysis should be improved in the following ways: AA, BB, CC*
Comment on strengths (as well as weaknesses) of the manuscript	*I commend the authors for their extensive data set, compiled over many years of detailed fieldwork. In addition, the manuscript is clearly written in professional, unambiguous language. If there is a weakness, it is in the statistical analysis (as I have noted above) which should be improved upon before Acceptance.*

Event detection in finance using hierarchical clustering algorithms on news and tweets

Salvatore Carta [1], **Sergio Consoli** [Corresp., 2], **Luca Piras** [1], **Alessandro Sebastian Podda** [1], **Diego Reforgiato Recupero** [1]

[1] Department of Mathematics and Computer Science, University of Cagliari, Cagliari, Italy

[2] Directorate A-Strategy, Work 9 Programme and Resources, Scientific Development Unit, Centre for Advanced Studies, European Commission, Joint Research Centre (DG-JRC), Ispra, Varese, Italy

Corresponding Author: Sergio Consoli
Email address: sergio.consoli@ec.europa.eu

In the current age of overwhelming information and massive production of textual data on the Web, Event Detection has become an increasingly important task in various application domains. Several research branches have been developed to tackle the problem from different perspectives, including Natural Language Processing and Big Data analysis, with the goal of providing valuable resources to support decision-making in a wide variety of fields. In this paper, we propose a real-time domain-specific clustering-based event-detection approach that integrates textual information coming, on one hand, from traditional newswires and, on the other hand, from microblogging platforms. The goal of the implemented pipeline is twofold: (i) providing insights to the user about the relevant events that are reported in the press on a daily basis; (ii) alerting the user about potentially important and impactful events, referred to as hot events, for some specific tasks or domains of interest. The algorithm identifies clusters of related news stories published by globally renowned press sources, which guarantee authoritative, noise-free information about current affairs; subsequently, the content extracted from microblogs is associated to the clusters in order to gain an assessment of the relevance of the event in the public opinion. To identify the events of a day d we create the lexicon by looking at news articles and stock data of previous days up to d

Event Detection in Finance Using Hierarchical Clustering Algorithms on News and Tweets

Salvatore Carta[1], Sergio Consoli[2], Luca Piras[1], Alessandro Sebastian Podda[1], and Diego Reforgiato Recupero[1]

[1]**Department of Mathematics and Computer Science, University of Cagliari, 09124 Cagliari (CA), Italy**
[2]**European Commission, Joint Research Centre (DG-JRC), Directorate A-Strategy, Work Programme and Resources, Scientific Development Unit, Via E. Fermi 2749, I-21027 Ispra (VA), Italy**

Corresponding author:
Sergio Consoli[2]

Email address: sergio.consoli@ec.europa.eu

ABSTRACT

In the current age of overwhelming information and massive production of textual data on the Web, Event Detection has become an increasingly important task in various application domains. Several research branches have been developed to tackle the problem from different perspectives, including Natural Language Processing and Big Data analysis, with the goal of providing valuable resources to support decision-making in a wide variety of fields. In this paper, we propose a real-time domain-specific clustering-based event-detection approach that integrates textual information coming, on one hand, from traditional newswires and, on the other hand, from microblogging platforms. The goal of the implemented pipeline is twofold: (i) providing insights to the user about the relevant events that are reported in the press on a daily basis; (ii) alerting the user about potentially important and impactful events, referred to as *hot* events, for some specific tasks or domains of interest. The algorithm identifies clusters of related news stories published by globally renowned press sources, which guarantee authoritative, noise-free information about current affairs; subsequently, the content extracted from microblogs is associated to the clusters in order to gain an assessment of the relevance of the event in the public opinion. To identify the events of a day d ~~we create~~ the lexicon by looking at news articles and stock data of previous days up to $d-1$.Although the approach can be extended to a variety of domains (e.g. politics, economy, sports), we hereby present a specific implementation in the financial sector. We validated our solution through a qualitative and quantitative evaluation, performed on the Dow Jones' *Data, News and Analytics* dataset, on a stream of messages extracted from the microblogging platform *Stocktwits*, and on the *Standard & Poor's 500* index time-series. The experiments demonstrate the effectiveness of our proposal in extracting meaningful information from real-world events and in spotting *hot* events in the financial sphere. An added value of the evaluation is given by the visual inspection of a selected number of significant real-world events, starting from the Brexit Referendum and reaching until the recent outbreak of the Covid-19 pandemic in early 2020.

1 INTRODUCTION

The outbreak and the rapid growth of modern digital sources for daily current affairs, such as online newswires, microblogging websites and social media platforms, have led to an overwhelming amount of information produced every day on the Web. For this reason, Event Detection has become increasingly important in the last two decades, since it allows users to disentangle this mass of scattered and, oftentimes, heterogeneous data, and to become aware of relevant world-wide facts in a more efficient way.

From a general perspective, an event, which is somewhere referred to as *topic* (Kumaran and Allan, 2004), can be defined as something that happens at a particular time and place (Allan et al., 1998b), causing a change in the volume of textual content that discusses the associated topic at a specific time

(Dou et al., 2012). In this sense, Event Detection aims to discover contents published on the Web that report on the same current topic, organize them in meaningful groups and provide insights, based on properties extracted automatically from the data (Allan et al., 1998b; Hu et al., 2017). It represents a valuable resource to create awareness and support decision making in various domains of application, including epidemics (Aramaki et al., 2011; Rosa et al., 2020), earthquakes (Sakaki et al., 2010), social events (Petkos et al., 2012) and economy (see Section 2.4), among others. In some cases, the scope of the event detection task is not limited to arranging the contents and providing analytics, but constitutes the basis for further algorithmic processing, like for example the development of automatic trading strategies in financial applications (Gilbert and Karahalios, 2010; Ruiz et al., 2012; Makrehchi et al., 2013).

Given the importance of Event Detection, an increasing number of researchers have focused their attention on this problem since the late 1990s, building on the theoretic foundations of Information Retrieval and, later on, taking advantage of the discoveries of Natural Language Processing, Text Mining and Big Data processing. Early works mainly based their approaches on traditional news stories as they started being digitalized (Allan et al., 1998b; Lam et al., 2001; Kumaran and Allan, 2004), while social media platforms like Twitter[1] and Stocktwits[2] have become the dominant data source in the last decade (Hasan et al., 2018; Atefeh and Khreich, 2015). However, it has been demonstrated by Petrovic et al. (2013) that Twitter still cannot replace traditional newswire providers when considering the coverage and the timeliness of breaking news. In fact, this study shows that, while Twitter has a better coverage of minor events ignored by other media, traditional newswire sources often report events before users on the social network. Another disadvantage of microblogs is that they contain a considerable amount of noise, such as irregular syntax, misspellings and non-standard use of the language, let alone the increasing phenomenon of *fake news*, which makes it difficult to extract valuable information (Kaufmann and Kalita, 2010; Ajao et al., 2018). In light of this, a promising line of research has provided evidence that combining multiple sources of information allows to mitigate the flaws and exploit the advantages of each medium, thus improving the quality of the event detection task (Musaev et al., 2014; Petkos et al., 2012; Thapen et al., 2016).

Inspired by these findings, we developed a domain-specific clustering-based event-detection method that exploits the integration of traditional news articles and Stocktwits messages (which from now on will be referred to as *tweets*, for simplicity) to identify real-world events and to generate alerts for highly relevant events on a daily basis. The main intuition behind the integration of traditional press and social media is that, even though the former represents an authoritative and noise-free source which is convenient to mine to get qualitative information, it fails, taken alone, to provide insights about the entity or the resonance of the events. On the contrary, microblogs contain a considerable amount of noisy and unreliable content, but have the advantage of reflecting the impact that events have on public opinion. Because of this, we decided to exploit traditional news articles to construct a qualitative basis for our event-detection approach and to integrate the social media data on top of that, in order to get a quantitative measure.

The proposed approach, which will be described in full detail in Section 3, is defined as domain-specific because it collects news from the same sphere of interest (e.g. economy, politics, sports) and represents these documents focusing on the words that are most relevant for that field. However, the approach can be applied to various domains with minimum modifications. For example, if we are interested in identifying events that brought happiness or sadness to people, one might use social media text elements instead of news and a sentiment index indicator created on the same interval time of the social text to associate each social post to its sentiment level. Thus the lexicon would consist of tokens used within social media posts and weighted depending on their sentiment indicators. Besides, please note that our approach performs real-time event detection as it is supposed to identify events of the day d without any knowledge of the future. In particular, it creates the lexicon by looking at news articles and stock data of previous days up to $d-1$ without looking at the future. In this paper, we present the implementation of the pipeline that we designed specifically for the financial domain, which is a field where Event Detection has had one of its most promising applications (see Section 2.4). Our motivation derives from the intuition offered by several works in the financial literature that, drawing inspiration from the *Adaptive Market Hypothesis* (Lo, 2004), show that public news have an impact on the stock markets, explaining a part of the return variance (Boudoukh et al., 2019). This justifies the need for automatic

[1] http://www.twitter.com
[2] http://www.stocktwits.com

tools that can support companies, traders and all the other actors involved in the market, providing an at-a-glance visualization of acquisitions, stock splits, dividend announcements and other relevant economic events (Hogenboom et al., 2013).

We validated our approach through an experimental evaluation based, on one hand, on the Dow Jones' *Data, News and Analytics* dataset[3], which contains news articles delivered by globally renown sources, and, on the other hand, on a set of messages collected from Stocktwits, a microblogging platform inspired by Twitter, where users posts short messages related to stock markets and trading. The events that constitute our ground truth for the alert generation algorithm were selected based on the stock price time series of the *Standard & Poor's 500* Index (S&P 500), following the intuition that relevant economic events lead to significant movements of the market. Our qualitative and quantitative analysis shows that the proposed method is able to extract meaningful, separable clusters, which correspond to real-world events. Furthermore, the alert generation algorithm detects *hot* events with high accuracy, proving the effectiveness of the integration of news articles and tweets.

The contributions of our work can be summarized as follows:

- we propose a novel approach to represent news documents exploiting a domain-specific lexicon created *ad-hoc* using the technique we have introduced in (Carta et al., 2020), where the lexicon has been generated on a different dataset;
- we design an original clustering-based event-detection approach that integrates news documents and tweets;
- we show the effectiveness of our method by means of an experimental evaluation performed on real-world datasets;
- we offer a visual inspection of the output obtained on a selected number of real-world events, including the Brexit Referendum, the U.S.-China trade war and the recent outbreak of the Covid-19 pandemic.

The reminder of this paper is organized as follows. Section 2 offers a thorough overview of the background research on Event Detection, analyzing works that deal with different kinds of media and application fields. The proposed approach is described in full detail in Section 3. The datasets and the methodology we have carried out for the evaluation are described in Section 4 while the obtained results are illustrated in Section 5. Finally, Section 6 contains general conclusions about this work and future lines of research.

2 RELATED WORK

The origins of Event Detection can be traced back to 1998, when a joint effort between the Defense Advanced Research Projects Agency (DARPA), the University of Massachusetts, Carnegie Mellon University and Dragon Systems aimed to define the problem within the wider field of Topic Detection and Tracking (TDT) and proposed an approach based on broadcast news stories that paved the way for new research on the field (Allan et al., 1998b,a; Yand et al., 1998). Since then, a considerable variety of algorithms have been proposed to tackle the problem, gradually taking advantage of the remarkable advances in Text Mining and Natural Language Processing. Most interestingly, the birth of social media platforms like Facebook, Twitter and Stocktwits in mid 2000s and their increasing popularity, together with the birth of the new era of Big Data (Marx, 2013), led to a widening of the range of data that could be exploited to detect real-world events. To note that it is common to employ lexicons for news representation for the financial domain. Within our previous work (Carta et al., 2020), we defined a strategy to generate industry-specific lexicons from news documents with the goal of dynamically capturing the correlation between words and stock price fluctuations. This has been then employed to solve a binary classification task with the goal of predicting the magnitude of future price changes for individual companies. Conversely, in this work we leverage the same technique to represent a new dataset and to solve a different problem, event detection.

In the following, we will hereby illustrate the previous research carried out in Event Detection, grouping it according to the type of source employed in the analysis - basically newswires, social media

[3] https://developer.dowjones.com/site/global/home/index.gsp

and an integration of heterogeneous data. Because the approach presented in this paper can be applied to different domains, also our overview of related works will cover a variety of fields, including health, security, sports and many others. However, we will conclude the overview by focusing on the financial sphere, since this is the specific domain within which our approach was developed and validated.

2.1 Newswires-based

The first type of data that has been explored in this field consists of traditional newswires and press releases, which, however, still have a primary role even in present research. Early works typically rely on *tf-idf* features to represent the documents in a Vector Space (Salton et al., 1975; Li et al., 2005) or Bag-of-Words (Zhang et al., 2010). Modification of these classic methods were proposed in order to enhance the representation by means of contextual information (Lam et al., 2001), lexical features (Stokes and Carthy, 2001), named entities (Kumaran and Allan, 2004), topic models (Yang et al., 2002) and, in more recent work, word-embeddings (Hu et al., 2017; Kusner et al., 2015). The most common approaches for the detection task are based on clustering, text classification or a combination of these (Atefeh and Khreich, 2015).

Going into more detail, authors in (Hu et al., 2017) exploit word-embeddings to overcome the downsides of *tf-idf* representation, namely sparsity and high dimensionality. On top of this, they build an adaptive online clustering algorithm that leads to an improvement in both efficiency and accuracy. Similarly, authors in (Zhou et al., 2018) enhance the *tf-idf* model by integrating the Jaccard Similarity coefficient, word-embeddings and temporal aspects of published news, with the goal of spotting *hot* events. Others (Mele et al., 2019) propose an algorithm to detect, track and predict events from multiple news streams, taking into account the publishing patterns of different sources and their timeliness in reporting the breaking news. They use a Hidden Markov Model (Beal et al., 2002) to represent current events and, subsequently, to predict facts that will be popular in the next time slice.

The main contribution of the proposed approach with respect to this line of research is the fact that, in our algorithm, the representation of the events extracted from news articles is enriched by the information mined on social media sources. In this way, we obtain a multifaceted perspective of events. Furthermore, another innovation regards the method employed to represent the textual data. In particular, our pipeline includes the creation of an *ad-hoc* lexical resource, which detects the words that are most relevant for a specific domain. During the construction of the vector representation of documents, only the word-embeddings of the selected terms are included, as described in full detail in Sections 3.2 and 3.3.

2.2 Social Media-based

Since the development of social media platforms and microblogging websites, a big share of the researchers' interest has been aimed at mining these sources of information for a more dynamic and multifaceted inspection of events. Among these platforms, the case of Twitter definitely stands out, becoming a *de facto* standard domain for Event Detection (Petrovic et al., 2013; Saeed et al., 2019). A thorough survey by Hasan et al. (2018), focused on Twitter-based approaches, suggests that this research branch can be split into three main categories: (i) methods that exploit properties in a tweet's keywords; (ii) methods that rely on probabilistic topic models; (iii) clustering-based methods.

For the first group, it is worth mentioning TwitInfo (Marcus et al., 2011), TwitterMonitor (Mathioudakis and Koudas, 2010) and EnBlogue (Alvanaki et al., 2011), which identify real-time trends on Twitter and allow the final user to browse large collections of messages, providing contextual information about tweets, visualizations and meaningful insights that describe the identified topics. Stilo and Velardi (2016) include temporal factors in their analysis in order to cope with the limited context of Twitter messages. Weng and Lee (2011) propose an approach that builds signals for individual words by applying wavelet analysis (Kaiser, 2010) on the frequency-based raw signals of the words; this method is able to spot the most relevant words and finally cluster them to form events.

Among the works that employ probabilistic topic models to represent tweets in a latent space, TwiCal (Ritter et al., 2012) is an open-domain event-extraction framework that identifies significant events based on a multitude of features including, but not limited to, contextual, dictionary and orthographic features. TopicSketch (Xie et al., 2016) is a system that identifies bursty topics from live tweet streams in an efficient way, by tracking the occurrence of word pairs and triples in small "sketches" of data. Zhou et al. (2015) devise a lexicon-based approach to spot tweets that are event-related and, based on these tweets, extract a structured representation of events by means of an unsupervised Bayesian model.

As for clustering-based approaches, Petrović et al. (2010) propose a time-efficient way to determine the novelty of a new tweet appearing in a live stream; novel tweets represent new stories and, therefore, will be assigned to newly created clusters, which are later ranked according to the number of unique user posts and the entropy information. The approach by Becker et al. (2011) groups tweets into semantically related clusters and then exploits a series of cluster properties (including temporal, social and topical features) in order to discriminate between real-world events and non-events messages. Analogously, Kaleel and Abhari (2015) employ a locality-sensitive-hashing scheme to extract clusters from the Twitter stream; the exploration of the clusters, which takes into account size, time and geolocation, leads to the identification of significant real-world events.

As already mentioned, the novelty of our approach with respect to these works is that social media data is not considered on its own, but in conjunction with news articles, in order to obtain a more insightful representation of events.

2.3 Integration of heterogeneous data

As stated in the Introduction section, several works in the literature suggest that, in many scenarios, an integration of different kinds of sources is necessary to improve the effectiveness of the event-detection algorithm, as far as both timeliness and coverage are concerned (Petrovic et al., 2013; Musaev et al., 2014; Petkos et al., 2012). As a consequence, a promising research branch has grown based on this principle. One interesting example is represented by the work by Osborne et al. (2012), which aims to mitigate the spuriousness intrinsic to Twitter messages by means of information from Wikipedia. The latter is used as a filter to discard large numbers of noisy tweets, thus refining the representation of the extracted events. Thapen et al. (2016) propose a methodology to automatically spot outbreaks of illness from spikes of activity in real-time Twitter streams. A summary of these events is provided to the user with the goal of creating situational awareness; this is achieved by presenting the most significant tweets and by linking them with relevant news, which are searched on the Web based on term occurrences. Petkos et al. (2012) develop a novel multimodal clustering algorithm to explore multimedia items extracted from several social media platforms, with the purpose of detecting social events. The authors suggest that the proposed approach can be extended to any scenario which requires the usage of multimodal data. In (Consoli et al., 2010, 2020) the authors present some novel optimization strategies for the quartet method of hierarchical clustering, a methodology popular in the context of biological phylogenesis construction by integration and clustering of different heterogeneous data.

Our approach differs from other works in this category in the way news and tweets are juxtaposed. In fact, the information extracted from news articles constitutes the basis of our event-detection algorithm, while the processing of tweets is implemented on top of that, with the goal of corroborating that information.

2.4 Event detection in Finance

Event detection, Natural Language Processing and Sentiment Analysis have been widely applied in the financial sphere to provide more and more insightful tools for supporting decision making (Xing et al., 2018). Some works have pushed the research as far as correlating the information about the events with the movement of the stock prices, with the goal of predicting future returns and developing trading strategies. Heston and Sinha (2017) study in which way the sentiment and the aggregation of the news affect the time horizon of the stock return predictability. In particular, through a neural network-based method, they show that daily news can forecast returns within one or two days, while aggregating news over one week provides predictability for up to 13 weeks. Moreover, the authors produce evidence that positive news stories increase stock returns quickly, while negative stories have a long delayed reaction. Schumaker and Chen (2009) combine news textual data and S&P 500 price time-series to estimate a discrete stock price twenty minutes after a news article was released, using Support Vector Machines (Suykens and Vandewalle, 1999). Ding et al. (2015) extract a structured representation of events from financial news, relying on the Open Information Extraction tool developed by Yates et al. (2007), and subsequently train a neural tensor network to learn event embeddings; this dense vector representation is then fed into a deep learning model to predict short-term and long-term stock price movements on S&P 500.

As far as social media-based approaches are concerned, Daniel et al. (2017) carry out an analysis of the content published on Twitter about the thirty companies that compose the Dow Jones Average. In particular, the authors start by detecting and discarding noisy tweets that might distort the information about relevant financial events; in the next steps, they perform a sentiment analysis on the valuable tweets

and correlate them with the behavior of the stock market. Authors in (Tsapeli et al., 2017) apply a bursty topic detection method on a stream of tweets related to finance or politics and, then, employ a classifier to identify significant events that influence the volatility of Greek and Spanish stock markets. Events are represented as feature vectors that encompass a rich variety of information, including their semantics and meta data. Starting from the same motivations, Makrehchi et al. (2013) collect a set of tweets related to companies of the S&P 500 index and label them based on the price movement of the corresponding stock. Then, they train a model on this set to make predictions on the labels of future tweets and, on top, create trading strategies that prove to give significant returns compared to baseline methods.

Another branch in financial event detection is focused on the extraction of potentially useful information, like events, from news and social media, that can represent a valuable resource for further algorithmic processing or for human-in-the-loop decision making. The Semantics-Based Pipeline for Economic Event Detection (SPEED) (Hogenboom et al., 2013) aims to extract financial events from news articles and annotate them with meta-data with an efficiency that allows real-time use. This is achieved through the integration of several resources, including ontologies, named entities and word disambiguators, and constitute a feedback loop which fosters future reuse of acquired knowledge in the event detection process. Jacobs et al. (2018) tackle the task of economic event detection by means of a supervised data-driven approach. They define the problem as a sentence level multilabel classification task, where the goal is to automatically assign the presence of a set of pre-determined economic event categories in a sentence of a news article. Following the same intuition, Ein-Dor et al. (2019) develop a supervised learning approach for identifying events related to a given company. For this purpose, the authors train a sentence-level classifier, which leverages labels automatically extracted from relevant Wikipedia sections.

Hogenboom et al. (2015) measured the effects of various news events on stock prices. They retrieved 2010 and 2011 ticker data and news events for different equities and identified the irregular events. Finally, they cleaned the ticker data of rare event-generated noise and obtained a dataset with a more accurate representation of the expected returns distribution.

Moreover, Nuij et al. (2014) presented a framework for automatic exploitation of news in stock trading strategies where events were extracted from news messages presented in free text without annotations. It turned out that the news variable was often included in the optimal trading rules, indicating the added value of news for predictive purposes.

The innovation that we bring with respect to the literature consists, first of all, in the integration of different sources to obtain richer representations of events. Second, we propose a method to estimate the resonance of an event based on the activity on social media platforms, and we leverage this measure to provide warnings to the final user. Last but not least, our method has been deployed for real-time detection of financial events, ~~although~~, within the evaluation we carried out we applied it on historical data but without ~~considering the future information of a day~~, under analysis.

3 PROPOSED APPROACH

The problem that we set out to tackle in this work is twofold. In the first place, we want to identify groups of news stories related to real-world events of a specific domain, on a daily basis. More precisely, given a day d and a look-back period of n days, our approach aims to extract k semantically related clusters made of text documents published by newswires providers during the n days before d. The parameter k is automatically estimated from the data so that it reflects the real events actually taking place in the best possible way. Each cluster is described by a set of properties, including relevant headlines and keywords, that are semantically correlated with the event represented by the cluster.

Secondly, we intend to tackle the problem of understanding whether a highly relevant event is taking place on a given day. Such an event is defined as *hot* and is associated with an increased amount of content published on a microblogging platform about that topic in the same time interval.

The main ideas underlying our proposed approach are the following:

- detecting the words that are more significant for the context under analysis can lead to more effective domain-aware representations of documents;
- clustering techniques allow to identify and distinguish events reported in news stories;
- the integration of social media data and news stories is key to spot *hot* events that are potentially noteworthy for the context under analysis.

In the following sections, we will describe the implementation of the algorithm that we designed for a specific scenario, namely the financial field. However, we would like to point out that our proposal can be generalized to any sphere of interest with minimum modifications, concerning mainly the filter applied to the news corpus and the numeric feedback used to assign a score to words in the lexicon generation phase.

3.1 Overall Architecture

The proposed algorithm is outlined in the pipeline in Figure 1, which is repeated for each single day d on which the event-detection task is executed. The first step consists of the generation of a dynamic, context-specific lexicon, which includes the list of words that have proven to have the biggest impact on the market in a given period before d (Carta et al., 2020). This resource is computed by combining two different data sources: on the one hand, words are extracted from financial news published in a time interval that typically ranges from 2 to 4 weeks previous to d. On the other hand, the stock price time-series of the chosen market is used to assign numeric scores to the words appearing in the press releases.

In the following we will give some formal notation to illustrate how this step corresponds to perform a marginal screening (Genovese et al., 2012), a form of variable selection which is proven to be more efficient than the Lasso and with good statistical accuracy. Let us assume that in the period $[d-\ell, d-1]$ the algorithm collects N articles, where a portion of them contains the term j. Then

$$f(j) = \frac{1}{N} * \sum_{1 \le k \le N} X_k(j) * \Delta_d(k),$$

where $X_k(j)$ is a dummy variable for whether term j appears in article k and $\Delta_d(k)$ is the return on the day d for article k. In this form, $f(j)$ is the slope of a cross-article regression of $\Delta_d = (\Delta_d(1), \ldots, \Delta_d(N))$ on the dummy variable $X(j) = (X_1(j), \ldots, X_N(j))$. More precisely, $f(j)$ are coefficients of a marginal regression. By sorting them by decreasing scores and selecting those whose values are over (under) some specified threshold, is similar to taking the first n and the last n. Moreover, in our lexicon construction, if S is the index set of positive and negative words (those corresponding to high or low stock variations), and $\hat{S} = \{j : f(j) \ge t^+ \text{ or } f(j) \le t^-\}$, under certain conditions $Prob(\hat{S} = S) = 1$ as N and the number of terms go to infinity. This corresponds to the sure screening property (Fan and Lv, 2008).

Once the specialized lexicon is obtained, it is applied as a filter on the news documents, so that only the terms that appear in the lexicon are retained. Subsequently, a document-embedding representation of each news story is constructed by computing the average of the word-embeddings of its filtered words.

After the news-modeling stage, the document-embeddings are fed to an agglomerative clustering algorithm, which returns a list of labels, which indicate the cluster to which each specific observation belongs, and a variable number of centroids. Intuitively, each cluster should correspond to the event discussed in the news contained in it, while the cluster centroid serves as an high-level discriminating representation of the event. The previous output is later refined through an operation of outlier removal, whose goal is to find and discard those documents whose assignment to their cluster is weak. Once the spurious data have been cleaned out from the clusters, a series of properties are extracted from each group of news, both for illustrative and for evaluation purposes. This information includes the titles of the articles, the percentage of positive and negative words (associated to high or low stock price variations, as described in the next paragraph), and the list of the most relevant words for the cluster, assessed through a *tf-idf*-based method.

At this point of the pipeline, the integration between news stories and social media data takes place. The idea here is, first, to find tweets that are semantically correlated to some group of news and, second, to detect if an event reported in the news has a wide resonance on the social media platform. More specifically, the tweets relevant for the market under analysis published on the most recent day of the time interval are collected and then represented with the same embedding-based method previously employed for the news. The assignment task consists of attaching every tweet to the closest news-cluster, according to a similarity measure calculated between the tweet-embedding and each news-centroid, as long as this distance is smaller than a defined *tweet distance threshold*; otherwise, the tweet is discarded.

The last step in the event-detection pipeline is the alert generation. This happens when the percentage of the assigned tweets w.r.t the overall number of tweets published on the most recent day of the time interval is bigger than a given *alert threshold*. In fact, this suggests that a considerable number of people on the social media platform are discussing some events reported in the news.

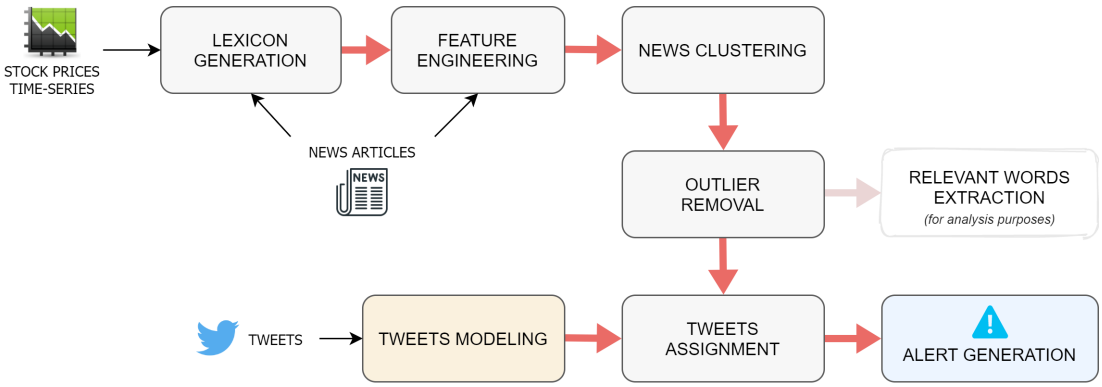


Figure 1. Overall architecture of the proposed approach.

3.2 Lexicon Generation

The lexicon generation stage leverages the method that we proposed in (Carta et al., 2020), which we hereby set out to illustrate for the sake of completeness. From a general perspective, the goal of the lexicon generation is to select the set of words that are most relevant for a specific domain in a given time interval. In order to be able to capture the impact of events that occur day by day (and thus the effect of new words that show up in news articles reporting such events), we perform the lexicon creation in a dynamic way, repeating its generation every day. For these reasons, we define the lexicons generated by our approach as *time-aware* and *domain-specific*.

If we apply this concept to the financial sphere, the relevance of a word can be estimated by observing the effect that it has on the market after the delivery of the news stories containing this word. In this sense, the resulting lexicons will capture potential correlations between words that appear in news stories and stock price movements: terms that are consistently followed by significant positive (negative) variations will receive a high (low) score, while terms that are followed by negligible or arbitrary variations will tend to have a score close to 0. Going into more detail, for each day we collect all the news that are relevant for the S&P 500 Index published during the time frame $[d-\ell,\ d-1]$ (with $\ell \geq 1$). More precisely, we select all news with at least one mention of *Standard & Poor* (or strictly related keywords like *SP500* and *SPX*). For each news article in this set, we extracted the text, consisting of the title, the snippet and the full body of the article, and then we performed some standard pre-processing techniques on it, such as stop-words removal (using that of Stanford CoreNLP[4]), stemming and tokenization (the last two using NLTK[5]). In addition, we removed from the corpus all the words that appeared too frequently and too infrequently, according to given tolerance thresholds. In our case, we filtered out all the words that appear in more than 90% of the documents or in less than 10 documents (both thresholds were set experimentally). Subsequently, we construct a document-term matrix, in which each row corresponds to a news article and date and each column corresponds to a term, as obtained after the pre-processing. In the next step, we iterate over the rows of the matrix and, for each of them, we assign to each of its terms a value equal to the stock price variation registered on the day after the article was published, defined as:

$$\Delta_{d'} = \frac{close_{d'} - close_{(d'-1)}}{close_{(d'-1)}}, \tag{1}$$

where $d' \in [d-\ell,\ d-1]$ is the day after the publication of the underlying article, and $close_{d'}$ is the price of the stock at the closing time of the market on day d'. Finally, each column is averaged (counting only non-zero entries), thus obtaining a list of terms, each associated to a score given by the average of the values assigned to them. We sort the terms by decreasing scores and select the first n and the last n. These are the ones associated to higher price variations, respectively positive and negative, and represent the time-aware, domain-specific lexicon that will be exploited for the news modeling phase.

[4] https://tinyurl.com/yygyo6wk
[5] https://www.nltk.org/

3.3 Feature Engineering

The aim of the news modeling phase is to obtain a representation of the news documents in a vector space, such that it captures its semantics and it is convenient for the subsequent cluster analysis. This must be achieved by characterizing each article through the words that are more relevant for the specific domain, ignoring the words that represent noise or that, in any case, do not provide valuable information. The two main resources that are exploited in this stage are the lexicons described in the previous Section 3.2 and a word-embedding model, which assigns a dense vector representation to words (Mikolov et al., 2013). The latter can be obtained by training the model on the text corpus under analysis or by loading a pre-trained model previously fit on an independent corpus.

First of all, each news article undergoes a series of standard text pre-processing operations, namely tokenization, conversion to lower case and stopwords removal. Subsequently, the words of each document are filtered by means of the lexicon produced on the day of the publication of the news, so that only the words that appear in the lexicon are retained. Finally, the word-embeddings of the filtered words are extracted and their average is computed to obtain the news-embedding.

3.4 News clustering

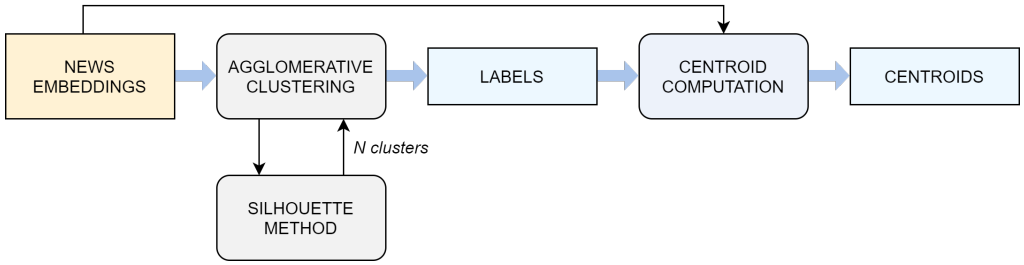


Figure 2. Flowchart of the clustering algorithm.

The embedding representation of news documents obtained in the previous step is the input to the clustering algorithm (Figure 2), whose goal is to split the articles in semantically-correlated groups. Ideally, each cluster corresponds to a real-word event.

For this purpose, we employ the *agglomerative clustering* algorithm. The decision mainly arises from a comparison with other standard techniques, which in this specific scenario do not prove as effective at separating the input data (see Section 4 for a detailed comparative analysis). The agglomerative clustering is a method pertaining to the family of hierarchical algorithms, which build nested clusters by merging or splitting them successively (Rokach and Maimon, 2005; Murtagh, 1983; Zhao et al., 2005). More specifically, the agglomerative algorithm follows a bottom up approach: at the beginning, each sample represents a cluster on its own, and clusters are successively merged together according to a linkage criteria. In this study, the choice fell on the average linkage criterion, which minimizes the average of the distances between all observations of pairs of clusters, while the affinity used to compute the linkage was the cosine distance, the most commonly employed metric when dealing with text documents.

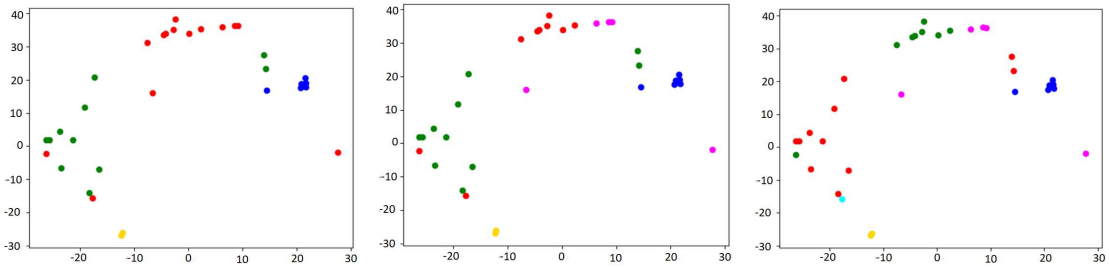
An important aspect to take into account is the number of clusters k that the algorithm extracts. This can be set as a parameter to the agglomerative method, but finding the most suitable k a priori is not trivial. Above all, using a fixed k for all days would in most cases lead to a misshaped approximation of reality, because the number of events taking place around the world naturally varies enormously from day to day and in different periods of the year. For this reason, a technique known as the *silhouette maximization method* is used to find the ideal value of k in a dynamic manner. The silhouette coefficient is a metric used to evaluate the performance of a clustering algorithm when a ground truth is not available. It ranges from -1 to 1, where higher scores relate to models with better defined clusters and it is defined for each sample by the following formula:

$$silhouette = \frac{(b-a)}{max(a,b)},$$

where a is the mean distance between a sample and all other points in the same class and b is the mean distance between a sample and all other points in the *next nearest cluster*. A global score for the

whole model can be easily computed as the average of all the scores computed on the single samples. In fact, the average silhouette coefficient is the metric that guides us in the choice of the best number of clusters k on each day on which the event-detection pipeline is executed. The agglomerative clustering algorithm is run with k values ranging from 2 to 10 and the silhouette score is computed on the output for every k. The value of k which led to the highest silhouette is selected.

Figure 3 illustrates the output of a small instance of the silhouette maximization method applied on a set of news collected in one week.



(a) Number of clusters: 4
Silhouette score: 0.24

(b) Number of clusters: 5
Silhouette score: 0.27

(c) Number of clusters: 6
Silhouette score: 0.24

Figure 3. Illustration of the silhouette maximization method. For space reasons, only the output with 4, 5 and 6 clusters is showed (Figures 3a, 3b, 3c, respectively). In this case, the algorithm would choose the number of clusters $k = 5$, which is the value that leads to the highest silhouette score (0.27 against 0.24 in the other two settings). The bi-dimensional visualization of news clusters is obtained by means of t-SNE, a tool to visualize high-dimensional data (van der Maaten and Hinton, 2008), which reduces the dimension of embeddings from 300 to 2. Every point represents a news in the 2D space and each color represents a different cluster.

The output of the agglomerative algorithm is simply a sequence of labels, which indicate the cluster to which each specific observation belongs. The method by itself does not return any centroid, as this notion is not employed in any step of its procedure. However, the next phases in the event-detection pipeline require also a centroid for each cluster (i.e., a vector obtained through a combination of the samples in the cluster, typically the mean or median). For this reason, we manually construct a series of centroids, computed as the median of the document-embeddings contained in the respective cluster. In this scenario, the median is a more suited choice compared to the mean, because it is less sensitive to noise and outliers. The resulting centroids, which are vectors of the same length of the document-embeddings, serve as high-level discriminating representations of the corresponding events.

3.5 Outlier Removal

Not necessarily all the articles published by press sources report events currently taking place: for example, in some cases they might refer to anniversaries of past happenings or they might discuss current affairs from a general perspective, including more than one event. This can cause noise in the formation of the clusters and, to some extent, can negatively influence the features of the centroid. For this reason, it is recommendable to detect and remove the outlier documents within each cluster (Figure 4). Intuitively, these are the observations on which the clustering algorithm was least effective.

Again, the silhouette coefficient (this time in its per-sample version) is used to spot the documents that were poorly clusterized: those with lower silhouette scores are typically the ones that lie on the border between two or more groups, causing a higher uncertainty in the clustering task. This is not enough, though: in fact, there might be samples that, even if they are not located on a border, have a weak correlation with the other articles of the same cluster: these are typically the documents that lie further away from the centroid of the group to which they belong. Therefore, the noise-reduction task that we designed exploits two different metrics in order to detect the outliers: the per-sample silhouette coefficient and the cosine distance from the centroid. First of all, the samples are sorted in decreasing order according to these two metrics, respectively, thus obtaining two different rankings. Then, *cutoff threshold* is defined on each ranking, by picking a percentile value computed on each of the two lists, respectively (typically somewhere between the 10th and the 30th). Finally, all the samples whose scores

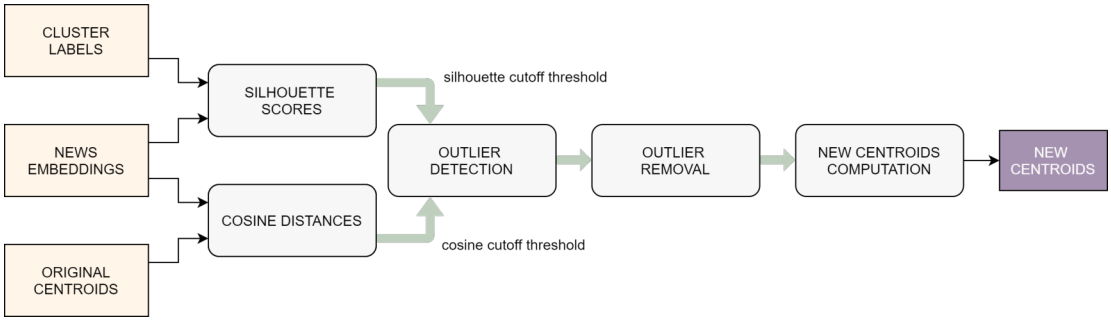
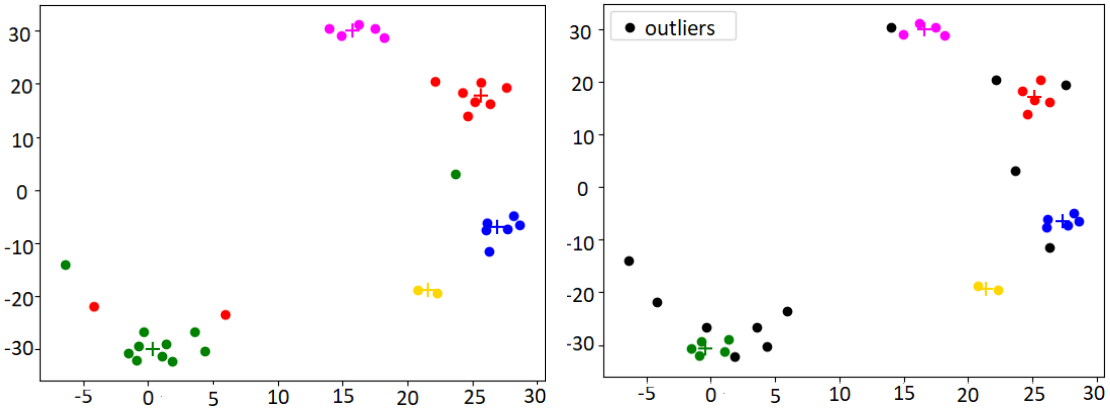


Figure 4. Flowchart of the outlier removal procedure.

are below the *cutoff threshold* in one of the two rankings are marked as outliers and suppressed. It is straightforward to note that choosing higher percentiles to set the *cutoff threshold* will make the algorithm more selective, in the sense that it will consider more documents as outliers. In rare extreme cases, this might lead to the total suppression of one of more clusters, if these already contained few samples in the first place and were not compact.

At this point, the new centroids of the affected clusters need to be computed, to account for the elimination of some of the documents. Similarly to what was done before, each centroid is obtained as the median of the document-embeddings that remain in the cluster after the outlier removal.

An example of clustering and outlier removal can be observed in Figure 5, which presents a bi-dimensional visualization of the clusters obtained from the financial news published on the U.S. press on the week before the Brexit referendum, an event that we will use along the paper that took place on the 23rd of June 2016.



(a) Original clustering **(b)** Clustering after outlier removal

Figure 5. Illustration of the outlier removal method on the weeks published in the week before the Brexit referendum. Figure 5a shows the original clusters including all documents. In Figure 5b the outliers are marked in black. For this example, the 30th percentile was used as the cut-off threshold, in order to make the effects of the algorithm more visible. Centroids are indicated by ”+” marks, in the same color of the respective cluster.

3.6 Relevant Words Extraction

There are several properties that can be extracted from each cluster to provide insightful information to the user. At the same time, they can serve as a useful means to perform a qualitative evaluation of the clusters, as they allow judging at first sight if the clusters are meaningful and coherent. These properties include the titles and snippets of the news articles, the time-span covered by the cluster, the percentage of positive and negative words from the specialized lexicon and the list of relevant words. Hereby we focus our attention on the latter (Figure 6), as the other ones are trivial to extract.

At first, all the news articles included in the current time interval are fed to a *tf-idf* model, regardless of their cluster. The features used to fit the model are the words included in the specialized lexicon, so this is equivalent to filtering the documents' words with the lexicon. The output of the model is a sequence of vectors, one for each document, where the values represent the relevance of the corresponding words for the document. At this point, the *tf-idf* vectors are grouped up according to the cluster to which the respective documents have been assigned. Then the average of the vectors is computed for each group, thus obtaining a unique score for each feature for each cluster, indicating the relevance of that word for that cluster. Finally, it is sufficient to rank the features and select the top n to get the list of the most relevant words for each group of news.

Tables 1 and 2 show the instance of the 3 most relevant headlines and the lists of the 10 most relevant words, respectively, for the clusters obtained on the day of the Brexit referendum. It is clear from the news titles that cluster#0 effectively captures this event, while the others are more focused on different aspects of business and finance (cluster#1 deals with global finance, cluster#2 with stock markets, cluster#3 with volatility, cluster#4 with the Federal Reserve system). The fact that several mentions of Brexit also appear in the headlines of the other clusters is attributable to the huge impact of the British referendum on many spheres of economy around the world. Not surprisingly, also the titles of the Brexit-cluster are characterized by the financial jargon, since the whole set of news on which the event-detection task is performed was selected by this specific field of interest. For the same reason, the variety of semantic fields involved in the lists of relevant words is not so wide between clusters. Noticeably though, these lists reflect quite accurately the content of headlines of Table 1.

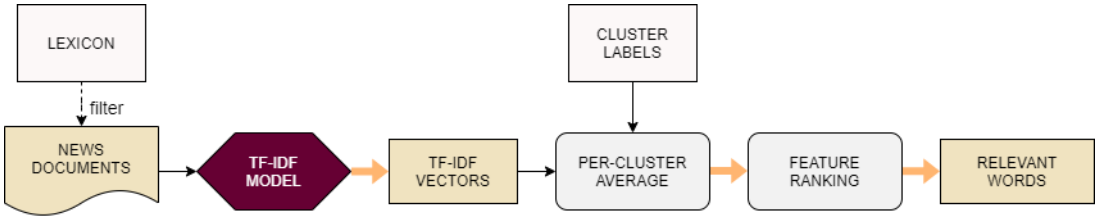


Figure 6. Flowchart of the algorithm used to extract the most relevant words from each cluster.

cluster	top-3 titles
#0	*- Markets Rise on U.K. Polls — Jumpy investors shift their bets as opinion surveys tilt slightly to Britain staying in EU* *- Relief Rally Lifts Stocks and Oil — Dow industrials gain 129.71 as bets rise that U.K. would stay in EU; crude jumps 2.9%* *- Global markets rally as polls show that enthusiasm for Brexit is waning*
#1	*- D.C. juggernaut in manufacturing is splitting in two* *- Global Finance: Abu Dhabi Banks Considering Merger — Deal would create biggest lender in Middle East; industry stocks rally in region* *- Global Finance: Bankruptcy Filing By Phone Firm Hits Big Brazilian Bank*
#2	*- As Fears of Brexit Ease, Wall St. Thrives* *- Health and Tech Shares Lead a Down Day for the Market* *- Market Ends a Losing Streak*
#3	*- This Time Around, the Volatility Index Matters* *- Stock Volatility Expected to Last*
#4	*- Stocks Fall 5th Day in Row — Fed rate decision likely means high-dividend shares will benefit as banks are pressured* *- Growth Tepid, Fed Slows Plan to Raise Rates* *- Brexit fears lead Fed to postpone increase in key interest rate*

Table 1. Lists of 3 most relevant titles (i.e., pertaining to the three documents that are closest to the respective centroids) for each of the 5 clusters obtained from the news collected in the week before the Brexit referendum (cluster#3 contains only 2 news documents in total).

clusters				
#0	**#1**	**#2**	**#3**	**#4**
polls	capital	cents	volatility	wednesday
eu	firm	gallon	options	economic
leave	business	yen	exchange	policy
stay	based	us	short	rise
british	protection	feet	indication	inflation
referenum	owns	cubic	matters	december
rising	majority	copper	rising	july
momentum	commercial	heating	historical	won
surveys	filing	wholesale	problem	results
volume	value	silver	worried	broader

Table 2. List of the 10 most relevant words for the cluster obtained on the day of the Brexit referendum.

3.7 Tweet Assignment

The goal of this phase is to enrich each cluster of news with a group of tweets that are semantically correlated with the event associated to the cluster (Figure 7). First of all, we collect from Stocktwits all the tweets relevant to the market under analysis, published on the most recent day of the time interval used for the event-detection task[6]. The duplicate tweets are removed in order to avoid the negative influence of spam. A vector representation is constructed for every tweet with the same method used for the news articles: the punctuation is removed, the text is tokenized, the words are filtered with the specialized lexicons and the average of the embeddings of the remaining words is computed.

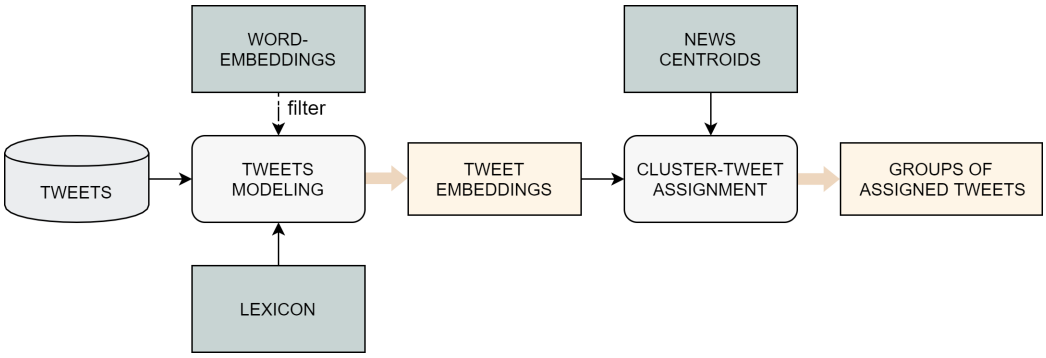


Figure 7. Flowchart of the tweet assignment task.

Subsequently, the actual assignment takes place. Each tweet-embedding is compared to each news-cluster centroid using the cosine similarity measure. The tweet is attached to the closest cluster only if this distance is smaller than a fixed *tweet distance threshold*; otherwise, the tweet is considered as noise and is not associated to any cluster.

An example of tweet assignment can be observed in Table 3, which presents the lists of the 3 most relevant tweets for the clusters obtained on the day of the Brexit referendum. Most importantly, the content of these tweets is totally coherent with the titles reported in Table 1. This means that the association of tweets to news-clusters was successful. It is noteworthy that even URLs, in case they contain meaningful keywords, can provide precious information for the semantic representation of the tweets and for the assignment task. This can be observed in the URLs of the first two tweets of cluster#0, which contain direct references to the Brexit referendum.

3.8 Alert Generation

The last step in the pipeline consists of the detection of the *hot* events: these are facts that not only have been reported in the news, but are also widely discussed on the social media platforms. The amount of

[6]This can be easily done by using the *cashtag* functionality, i.e. by searching for tweets that contain the symbol $ followed by the market code.

CLUSTER	TOP-3 ASSIGNED TWEETS
#0	*The polls are closer than the establishment cares to admit* *http://uk.reuters.com/article/uk-britain-eu-tns-poll* *Cameron and Osborne have credibility issues with British* *http://www.express.co.uk/news/uk/682561/david-cameron-eu-referendum-european-union-brexit-germany-boris-johnson-brussels* *EU referendum outcomes explained* *https://www.youtube.com/watch?v=VRIF4C_c2qs*
#1	No tweets assigned.
#2	No tweets assigned.
#3	*Gotta love this crazy volatility market* *S&P 500 squeeze back to 208 #volatility* *Fundamentals Still Look Solid Despite Brexit-Induced Volatility*
#4	No tweets assigned.

Table 3. List of the 3 most relevant tweets (i.e. closest to the respective centroid) for each of the 5 clusters obtained from the news collected in the week before the Brexit referendum.

content produced on the Internet about a certain episode is an insightful indicator of the entity of that episode and its consequences. For example, a remarkable popularity of a certain event among the users of Stocktwits is likely to translate into a potential impact on the market, since this website deals mainly with business and finance. Hence the importance of generating alerts that make the investor or trader aware of factors that they should take into account before operating on the market.

This task exploits the tweets-cluster assignment produced in the previous step and simply checks if the percentage of assigned tweets (among all clusters) with respect to the overall number of tweets published on the most recent day of the time interval (thus including also the discarded tweets) is above a fixed *alert threshold*. If this is true, an alert is generated.

The plot in Figure 8 shows the percentage of assigned tweets between the 10th of June 2016 and the 10th of July 2016. As expected, a peak is observed on the 23rd of June, day of the Brexit referendum, and an alert is generated.

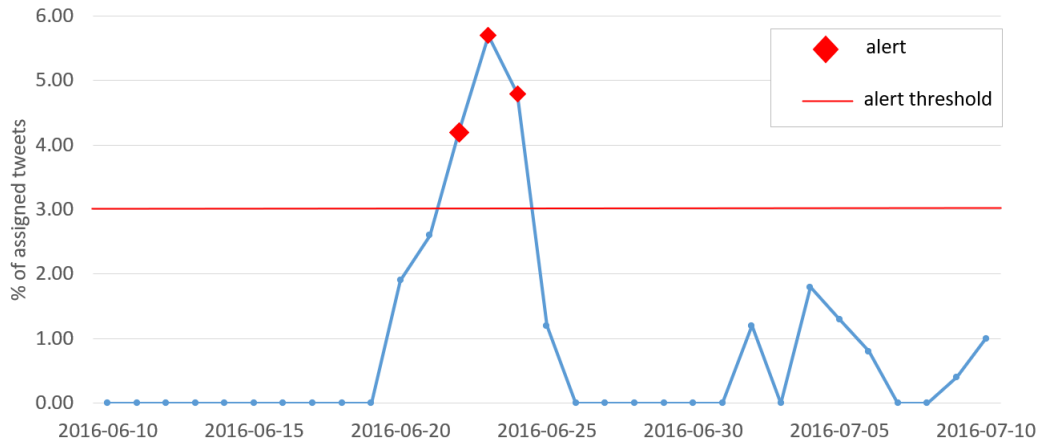


Figure 8. Plot of the percentage of assigned tweets (among all clusters) with respect to the overall number of published tweets, for each day in the interval between the 10th of June 2016 and the 10th of July 2016. The red markers indicate the generated alerts, while the red horizontal line represents the *alert threshold*.

4 EXPERIMENTAL SETTINGS

In this section we will illustrate the datasets we have employed within our study and the methodology we have followed for the experimental evaluation.

4.1 Datasets

Dow Jones DNA

The Dow Jones "Data, News and Analytics" dataset[7] provides documents from more than 33,000 globally renowned newspapers, including e.g. *The Wall Street Journal*, the *Dow Jones Newswires* and *The Washington Post*. The publications are both in print and online format and cover a wide variety of topics, such as finance, business, current affairs and lifestyle. The delivery frequency ranges from ultra-low latency newswires to daily, weekly, or monthly editions. For every article in the dataset, the headline, the snippet and the full body are available. Furthermore, every item is enriched with a set of metadata providing information about the source, the time and place of the publication, the relevant companies and the topics, among others.

Content usage rights vary based on the specific content, API, or feed combination. These rights include the display for human consumption or text mining for machine consumption and the content retention period.

Stocktwits data

Stocktwits[8] is a social media platform designed for sharing ideas between investors, traders, and entrepreneurs. It was founded in 2008 and currently counts over two million registered community members and millions of monthly visitors. Inspired by Twitter, it allows users to share and explore streams of short messages with a maximum 140 characters, organized around tickers referring to specific stocks and financial securities in general. This is achieved through the use of *cashtags*, which consists of the symbol "$" followed by the code of a financial security (e.g., "$AAPL", "$FB").

The dataset that we employed in our study contains the entire stream of tweets about S&P 500 published between June 2016 and March 2020. These messages were downloaded by means of the official API[9], selecting only the ones that contained the cashtag "$SPX", which corresponds to the aforementioned stock. The whole obtained collection contains 283,473 tweets.

Beside the full text of the tweet, every item in the dataset comes with a set of metadata, including the exact time of the publication, the number of "likes" (positive reactions by other users) received by the tweet, the sentiment score associated with the content and the number of the author's followers.

Standard & Poor's time-series

Another fundamental data source exploited in our analysis consists of the stock price *time series* of the the Standard & Poor's 500 Index, which measures the market performance of 500 large companies listed on stock exchanges in the United States. Companies are weighted in the index in proportion to their market value. The 10 largest companies in the index account for 26% of the market capitalization of the index. These are, in order of weighting, Apple Inc., Microsoft, Amazon.com, Alphabet Inc., Facebook, Johnson & Johnson, Berkshire Hathaway, Visa Inc., Procter & Gamble and JPMorgan Chase.

The dataset that we used for our evaluation was collected at a daily frequency and includes the following information:

- *open*: price of the stock at the opening time of the market;
- *close*: price of the stock at the closing time of the market;
- *high*: maximum price reached by the stock during the day;
- *low*: minimum price reached by the stock during the day;
- *volume*: number of operations performed on the stock during the day.

The aforementioned indicators are collected in an aggregated way, taking into account the values recorded for all companies included in the index.

[7] https://developer.dowjones.com/site/global/home/index.gsp
[8] http://www.stocktwits.com
[9] api.stocktwits.com/developers/docs

4.2 Methodology and Settings

The goal of the experimental framework that we designed to evaluate the proposed approach is twofold: on the one hand, we wish to verify that the clustering algorithm, supported by the news-modeling method and the outlier removal, is effective at separating the news stories according to their content and, thus, at providing an insightful way to inspect events. On the other hand, we want to assess the accuracy of the alert-generation algorithm, in particular to confirm that there is a correlation between *hot* events spotted by our approach and remarkable real-world events. For our purposes, we performed the following set of experiments: i) comparison of different clustering techniques; ii) event-detection qualitative evaluation; iii) alert-generation assessment.

Assessing the performance of an event-detection task is a daunting task, and getting a thorough quantitative evaluation is not trivial as well. This is partly due to a certain degree of subjectivity implied in the definition of an *event*, even more when we consider it with respect to a very specific context. In fact, an important occurrence such as a terrorist attack taking place in Europe, which is relevant in an absolute sense, might not be perceived as a relevant event in the sphere of U.S. finance. Moreover, to the best of our knowledge, universally recognized benchmarks of financial events are not available. For these reasons, in order to limit the subjectivity of the choice, we decided to select a list of events in a deterministic way, based on the weekly variations of the S&P 500 Index (more details on the selection method will be given in Section 5). Intuitively, we follow the assumption that important financial events are commonly associated with significant reactions of the stock markets, as suggested by the plots in Figures 9,10 and 11, that show the correlation between the weekly variations of S&P 500 stock price and relevant events taking place in the U.S. and in the rest of the world.

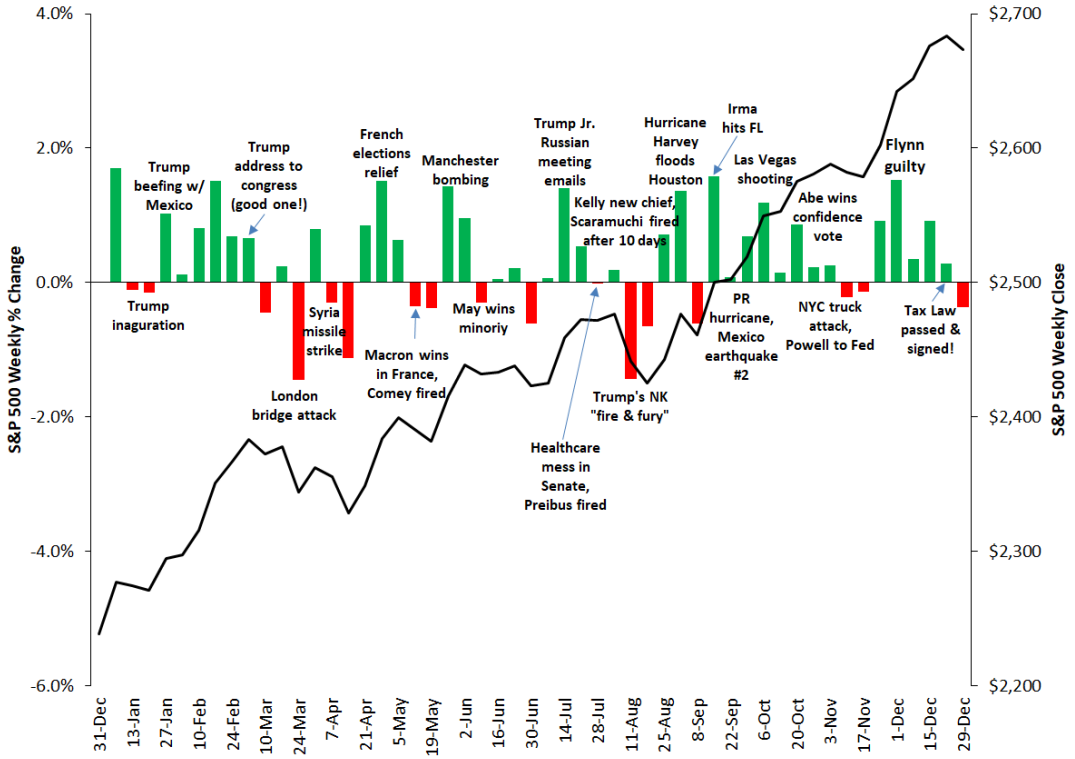


Figure 9. Correlation between the weekly variations of the S&P 500 stock price and relevant events taking place in the U.S. and worldwide in 2017. Source: `https://gordianadvisors.com/`

The datasets presented in Section 4.1 were filtered in order to extract the data most suited for our evaluation. Specifically, we selected from Dow Jones DNA all the news in English language published between June 2016 and March 2020, containing the keyword "Standard & Poor's" (as well as strictly related keyword like "S&P 500" or "SP500") in the title or in the body of the article. We aligned this collection of news with the Stocktwits data, which was collected in the same time-span, as already

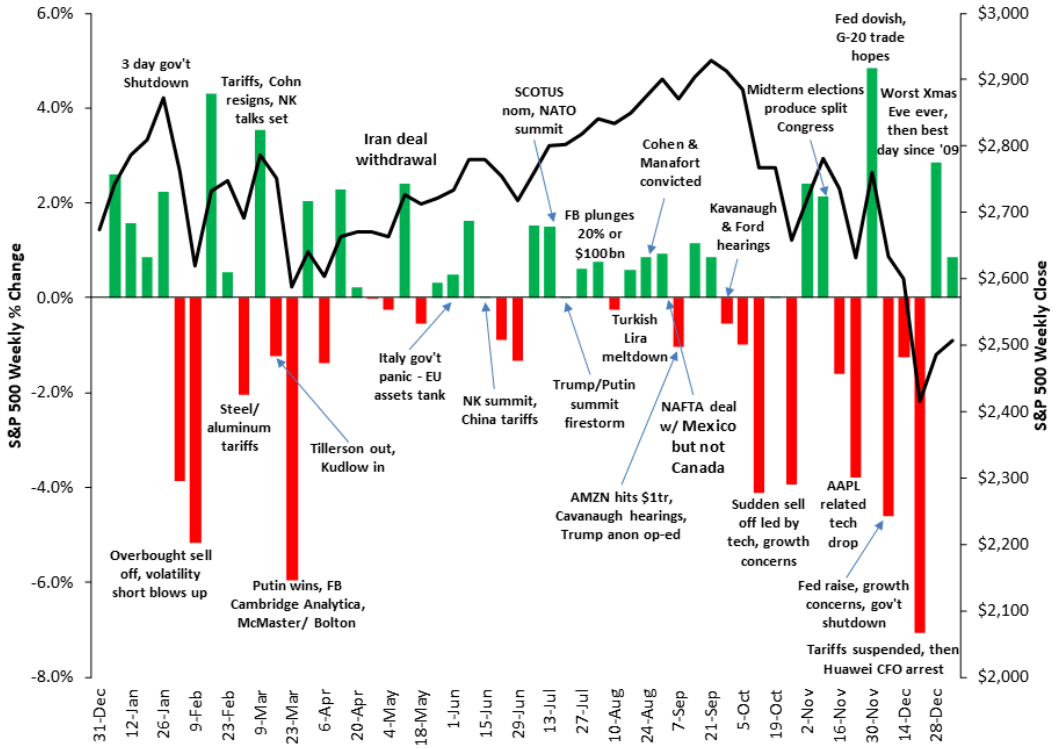


Figure 10. Correlation between the weekly variations of the SP500 stock price and relevant events taking place in the U.S. and worldwide in 2018. Source: `https://gordianadvisors.com/`

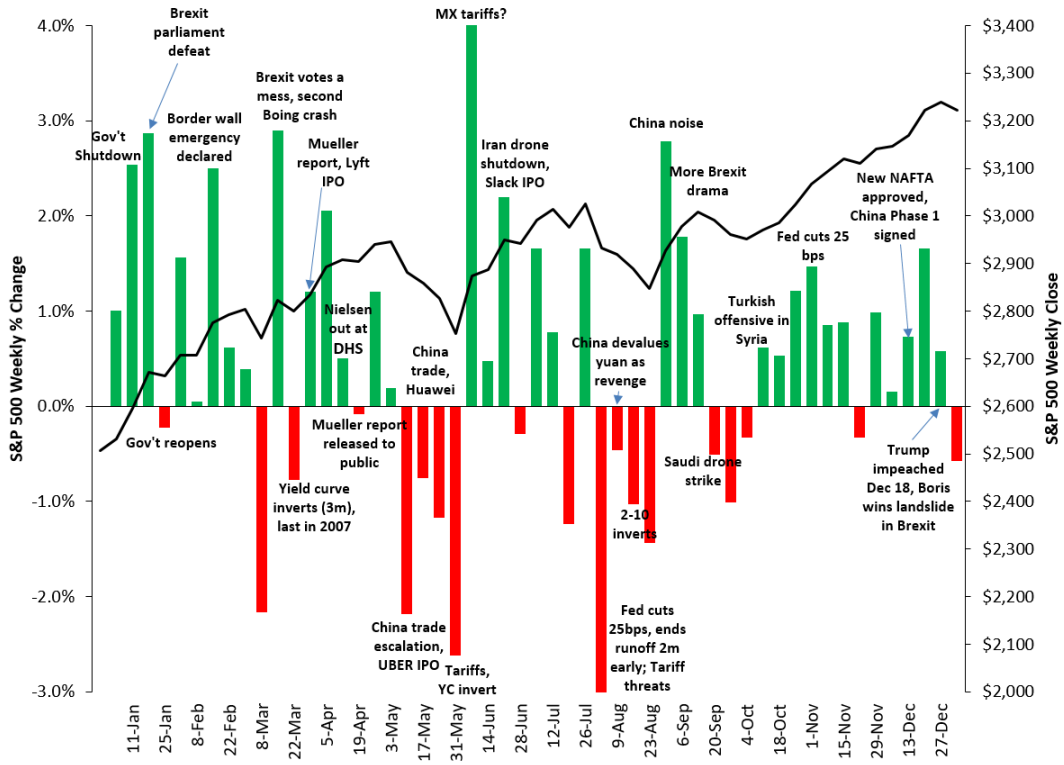


Figure 11. Correlation between the weekly variations of the SP500 stock price and relevant events taking place in the U.S. and worldwide in 2019. Source: `https://gordianadvisors.com/`

mentioned above. The whole filtered sets are thus composed of 8,403 news and 283,473 tweets. Finally, we considered the S&P 500 daily price time-series in the same years. The approach and experimental framework were developed in Python employing a set of open source Machine Learning libraries. The implementations of the agglomerative clustering algorithm, dimensionality reduction, TF-IDF method, and clustering performance metrics, were based on the popular scikit-learn library[10]; the K-Means, K-Medoids, K-Medians algorithms used for comparison were implemented through the Pyclustering library[11]; Natural Language Toolkit[12] and gensim[13] libraries were exploited for text pre-processing. As far as word-embeddings are concerned, we relied on the pre-computed word2vec model based on (Mikolov et al., 2013), trained on part of a Google News dataset composed of about 100 billion words. The model contains 300-dimensional vectors for 3 million words and phrases[14]. Throughout the experiments presented in the paper, the parameters were set as follows (if not specified otherwise): each daily lexicon was created on a set of news documents collected from a time window of 4 weeks, excluding all stopwords and terms that appeared in more than 90% or less than 10 documents, and the final lexicon consists of the words below the 20th and above the 80th percentiles of the ranking. The look-back window to collect the news documents to be clusterized on each day is 7 days. The *cutoff threshold* for the outlier removal stage is set to the 15th percentile. The *tweet distance threshold* for the tweet-assignment task is set to 0.5; the *alert threshold* is set to 3%. All the values of these parameters were carefully selected experimentally.

5 RESULTS

In this section we will show the results we have obtained. In particular we will show the results related to the clustering algorithm, those related to three specific events, and those related to the alert-generation algorithm.

Clustering performance evaluation

The first aspect we investigate is the choice of the clustering algorithm. As mentioned, the average Silhouette Coefficient is a standard metric to evaluate the goodness of a set of clusters. However, since it plays a role in the very construction of the clusters, we need some additional counter-checks to make the assessment more robust and less skewed. For this reason, we decided to include three more indicators in our evaluation:

- *Dunn Index*: similarly to the Silhouette Coefficient, it is a standard metric used to assess the performance of a clustering method when the ground truth is not available. It ranges from 0 to 1, with higher values indicating better clustering and is defined as:

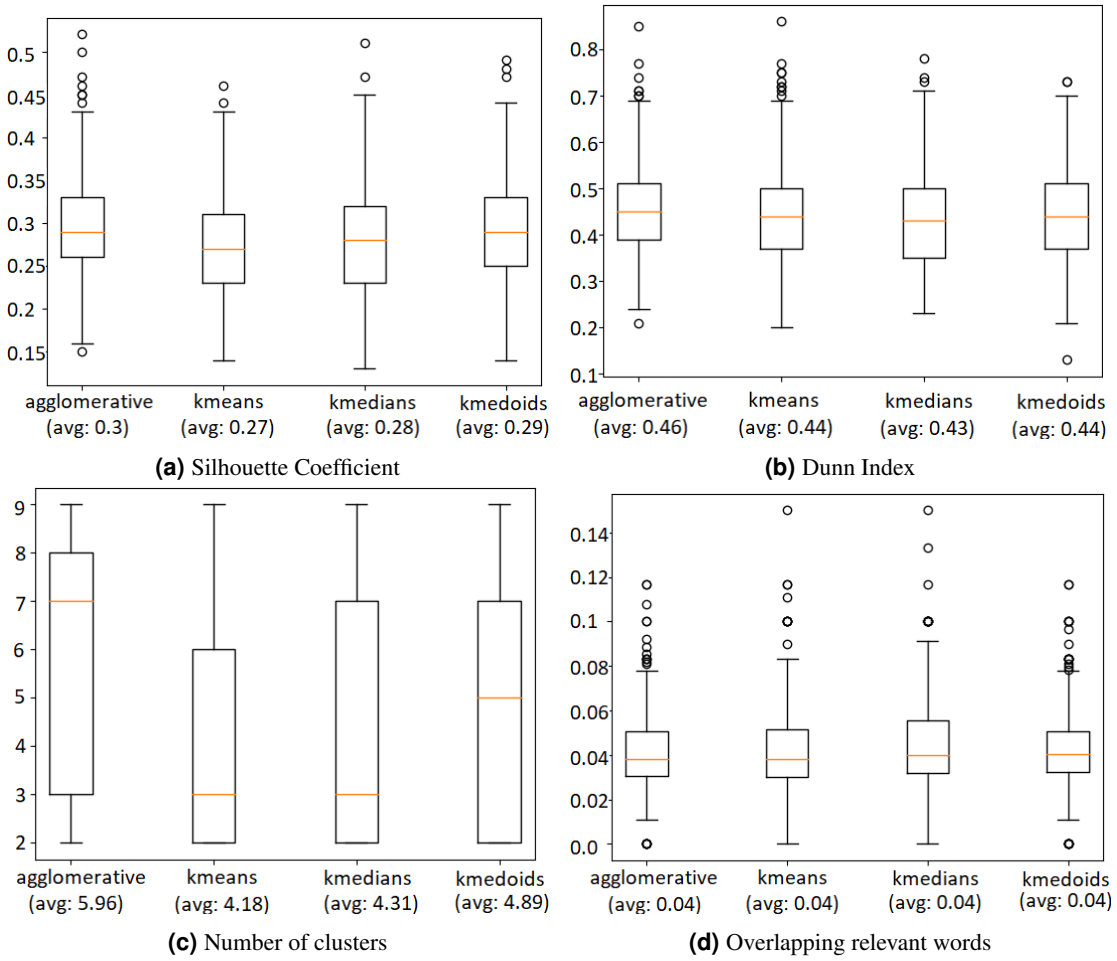
$$Dunn\ Index = \min_{1\leq i\leq c}\left\{\min_{i\leq j\leq c, i\neq j}\left\{\frac{\delta(X_i,X_j)}{\max_{i\leq k\leq c}\{\Delta(X_k)\}}\right\}\right\},$$

  where c is the total number of clusters, $\delta(X_i,X_j)$ is the intercluster distance between clusters X_i and X_j and $\Delta(X_k)$ is the intracluster distance within cluster X_k.

- *Number of extracted clusters*: this is also a useful indicator to evaluate the quality of a set of clusters, as higher values typically suggest a better separability of the data.

- *Overlapping between the clusters' relevant words*: it is estimated by computing the Jaccard Index[15] between the lists of top-10 relevant words for each pair of clusters, and by averaging the results. A small average overlapping signifies that news documents belonging to different groups discuss different topics and, therefore, that the articles were properly split according to their semantic content.

[10] http://scikit-learn.org
[11] http://pyclustering.github.io
[12] http://www.nltk.org
[13] http://radimrehurek.com/gensim/index.html
[14] http://code.google.com/archive/p/word2vec/
[15] The Jaccard Index between two lists is defined as the size of their intersection divided by the size of their union



(a) Silhouette Coefficient **(b)** Dunn Index

(c) Number of clusters **(d)** Overlapping relevant words

Figure 12. Comparison of the Silhouette Coefficient, Dunn Index, number of clusters obtained by different clustering algorithms and overlapping between the clusters' relevant words. The horizontal orange line represents the median of the obtained scores whereas the average is indicated between parenthesis. For further details we refer the reader to the official documentation of matplotlib library: `https://matplotlib.org/3.1.1/api/_as_gen/matplotlib.pyplot.boxplot.html`.

We used these metrics and the Silhouette Coefficient to compare four different techniques, namely Agglomerative clustering, K-Means, K-Medians and K-Medoids. These algorithms were executed on the same instances of the data selected for our evaluation, on each day of the time interval, using a look-back window of 1 week. Figure 12 shows the outcome of this experiment, indicating that Agglomerative is the algorithm that leads to better performance in terms of Silhouette, Dunn Index and most remarkably in the number of extracted clusters, while the overlapping of relevant words does not differ much. Please consider that the metrics were computed only after the outlier removal phase, which is responsible for an improvement of approximately 50% of both Silhouette and Dunn Index.

Event-detection evaluation

The results presented in the previous section, although obtained through an unsupervised validation, prove by themselves the effectiveness of the clustering algorithm at detecting events from a set of news. For illustration purposes, in this section we will carry out a qualitative analysis of the output of the clustering algorithm, focusing our attention on three specific events:

- The 2016 United States Presidential Elections (8th November 2016);
- The escalation of the U.S.-China trade war (9th May 2019);
- The outbreak of the Covid-19 pandemic (28th January 2020).

The reader notices that these three events are well-known world wide and there was no need to agree on those days. We invite the reader to refer to Section 3 for an analysis of the Brexit referendum in June 2016.

For each event, we picked one date among the most significant ones: the 8th of November is the actual date of the 2016 U.S. elections, which brought to the victory of Donald Trump[16]; the 9th of May 2019 is a few days after Trump threatened to raise tariffs on China and just one day before U.S. actually increased tariffs from 10% to 25%[17]; the 28th of January is the first day on which the total number of confirmed cases of Covid-19 worldwide surpassed the one-thousand threshold, passing from 793 to $1,786$ with a dramatic 125% daily change[18]. We applied the event-detection algorithm on the news published in the previous week, not including that date.

In Figure 13 we illustrate the results of the event-detection for each tested clustering algorithm. We remind that the Agglomerative clustering outperforms the others (as it can also be seen from the plots), and, therefore, we will focus our analysis on it (subfigures a, b and c).

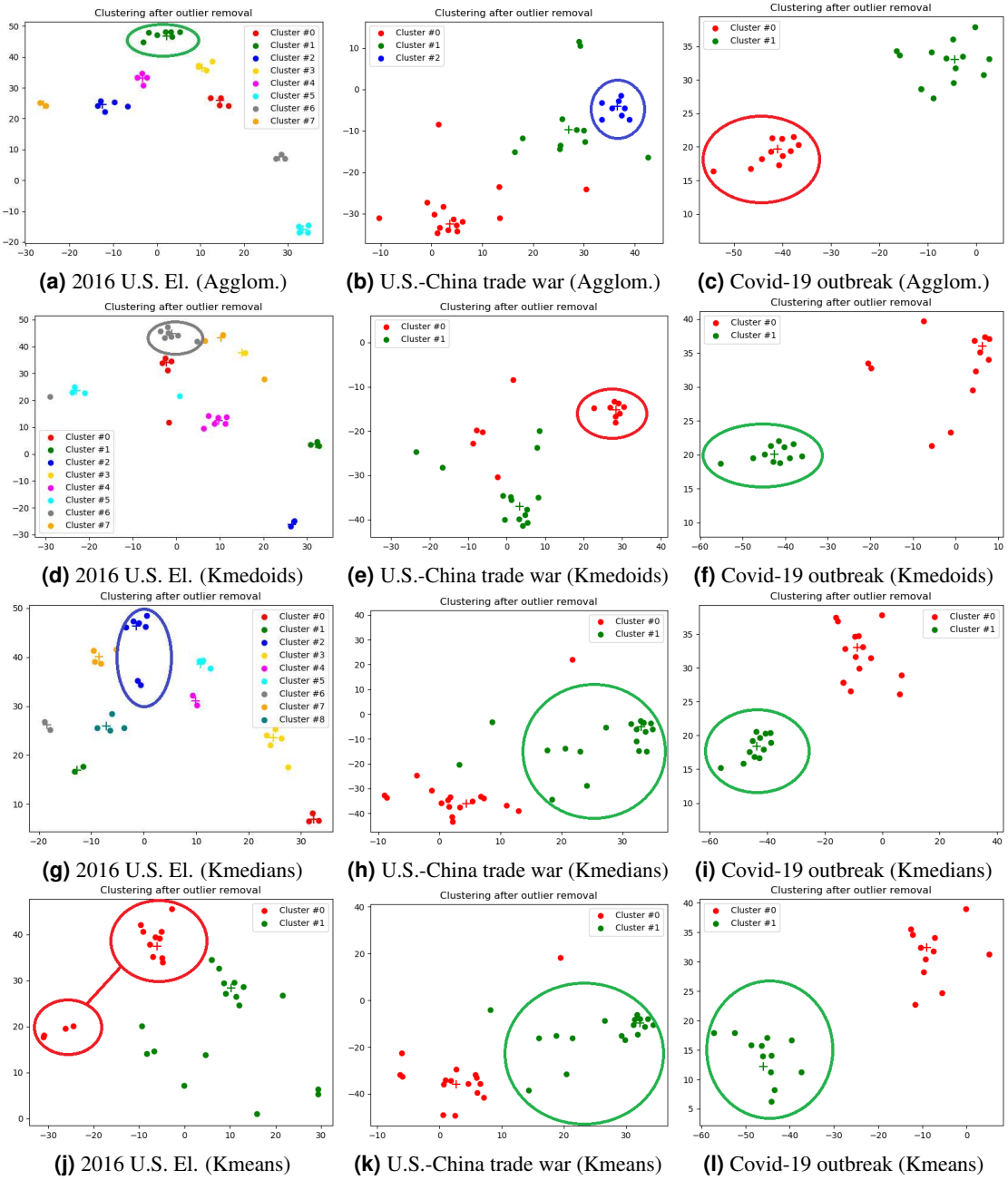
From the 2D visualizations presented in Figure 13, it can be seen that the points representing the news documents properly group up to form clusters. Interestingly enough, Figure 13c shows a strong polarization of the news, with only two clusters obtained. This is probably ascribable to the epochal impact of the Covid-19 outbreak, that drew a considerable part of the attention of the media, with many other topics left uncovered in the press. The average Silhouette Coefficient is decidedly above 0 for all three case studies (0.28, 0.27 and 0.36, respectively), indicating a satisfactory performance of the Agglomerative algorithm. These results are confirmed by the lists of relevant words (Table 4), relevant news headlines (Table 5) and relevant tweets (Table 6), which accurately reflect the semantics of the events (these last have been generated from the agglomerative clustering output only).

Alert-generation evaluation

As mentioned in Figure 1, the Alert-generation is the last step of the proposed pipeline and is performed on top of the clustering results and the tweets assignment to the generated clusters. The accuracy of the alert-generation algorithm can be gauged in terms of how many *hot* events it is able to spot in a given ground truth. As mentioned in Section 4.2, we selected the ground truth for our evaluation by looking at the weekly variations of the S&P 500 Index. More in detail, for every day d between June 2016 and March 2020 we compute the variation, in absolute value, between the close price of d and the close price of $d+7$ (i.e., 7 days after d). This quantity is formally defined as:

$$\Delta_d = \frac{|\, close_{(d+7)} - close_d \,|}{close_d}. \tag{2}$$

[16] http://en.wikipedia.org/wiki/2016_United_States_presidential_election
[17] http://china-briefing.com/news/the-us-china-trade-war-a-timeline
[18] http://covid19.who.int



(a) 2016 U.S. El. (Agglom.) **(b)** U.S.-China trade war (Agglom.) **(c)** Covid-19 outbreak (Agglom.)

(d) 2016 U.S. El. (Kmedoids) **(e)** U.S.-China trade war (Kmedoids) **(f)** Covid-19 outbreak (Kmedoids)

(g) 2016 U.S. El. (Kmedians) **(h)** U.S.-China trade war (Kmedians) **(i)** Covid-19 outbreak (Kmedians)

(j) 2016 U.S. El. (Kmeans) **(k)** U.S.-China trade war (Kmeans) **(l)** Covid-19 outbreak (Kmeans)

Figure 13. Illustration of the news clusters extracted on the three case studies considered in our qualitative analysis. The cluster associated to the event is highlighted by a circle. The correspondence between cluster and event is easily understood by manually reading the relevant words and the headlines of the documents that were associated to that label. For information about the 2D visualization technique, please refer to caption in Figure 3.

EVENTS		
2016 U.S. elections	**U.S.-China trade war**	**Covid-19 outbreak**
clinton	tariffs	virus
monday	percent	covid-19
trump	trump	outbreak
election	chinese	chinese
percent	united	losses
october	talks	impact
team	deal	europe
victory	friday	department
polls	imports	boeing
presidential	goods	hopes

Table 4. Lists of top-10 relevant words for the three case studies considered in our qualitative evaluation.

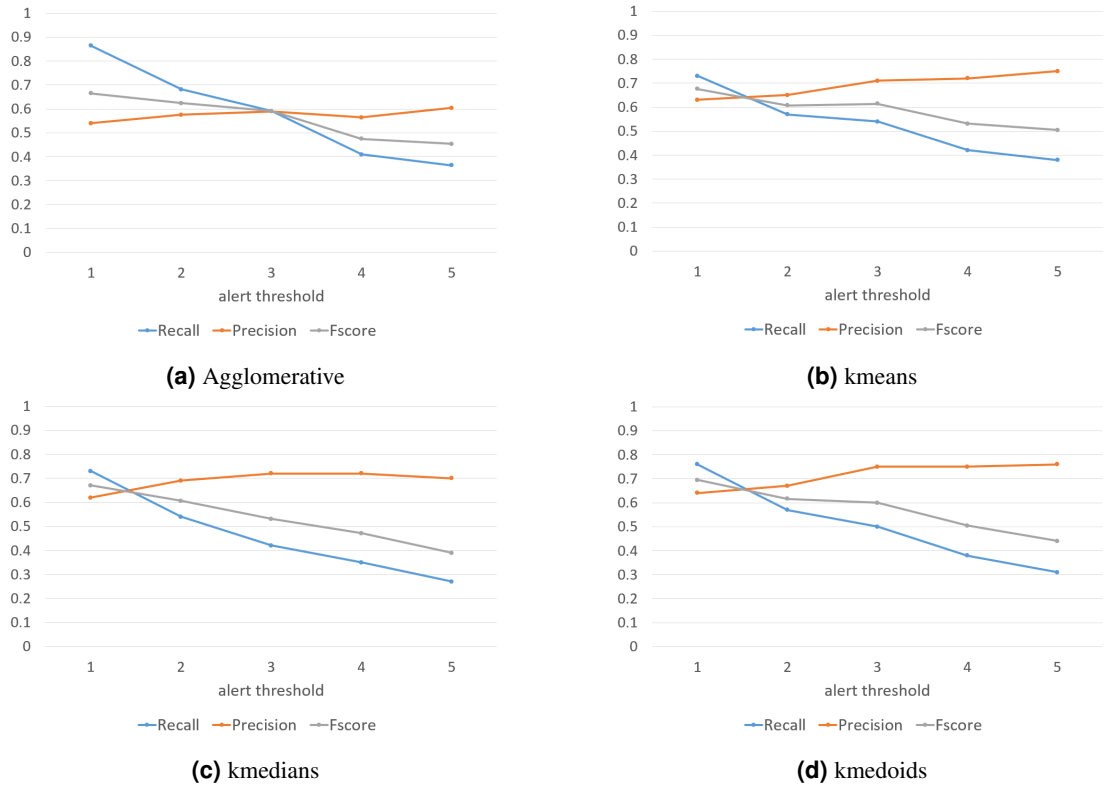
EVENT	TOP-3 RELEVANT HEADLINES
2016 U.S. elections	*- World Stocks: Dollar, Asia Stocks Rise on Clinton News* *- Stocks: Election Has Foreign Funds Wary - After Brexit surprise, some avoid investing in U.S. stocks until president determined* *- Election Presents Dilemma - Markets can't price in both a Trump win and a Democrat sweep of Congress at same time*
U.S-China trade war	*- Fear of Tariffs Jolts Markets And Nerves* *- U.S. Advisers Say China Is Reneging On Trade Accord* *- U.S. News: Tariffs Would Hit Consumer Goods*
Covid-19 outbreak	*- World markets show signs of fever* *- Markets on the slide as fears spread over virus* *- Britons returning from China to be 'safely isolated' for 14 days*

Table 5. List of the 3 most relevant headlines (i.e. closest to the respective centroid) for the three events considered in the qualitative evaluation.

EVENT	TOP-3 RELEVANT TWEETS
2016 U.S. elections	*- Hillary Clinton Wins!* *- The stock market's continual favoritism of Hillary Clinton proves that she has been bought. Corruption loves company.* *- Markets says "Hillary Clinton Wins". Congratulation New President*
U.S-China trade war	*- Goldman Sachs think the increase in tariffs will be narrowly avoided. Odds of new tariffs at 40% if the Chinese delegation still comes.* *- Tariff increase on Chinese imports will take effect on May 10 - Federal Register* *- "Reuters: Trump's punitive tariffs will burden consumers"; yeah like it...*
Covid-19 outbreak	*- Mainland Chinese, Hong Kong stocks tumble as Covid-19 death toll rises* *- Second U.S. Covid-19 case is Chicago resident who traveled to Wuhan* *- 3M Ceo says there factories are working 24/7 making masks & protective equipment to fight the virus. Buy your calls while there cheap. #stocks #covid-19*

Table 6. List of the 3 most relevant tweets (i.e. closest to the respective centroid) for the three events considered in the qualitative evaluation. Please keep in mind that the time interval used for the U.S. elections does not include the outcome of the polls (hence the wrong forecasts by users that initially proclaimed Hillary Clinton's victory).

The days d for which $\Delta_d > 0.02$ are marked as *event days*. The threshold of 0.02, which corresponds to a 2% weekly variation, is set experimentally and leads to approximately 15% of days being marked as an *event day*. Consecutive *event days* are aggregated to form events, which are thus defined as contiguous intervals, delimited by a *start date* and an *end date*. In order to avoid splitting intervals that actually refer to the same real event, we ignore interruptions of up to 3 days in the chain of consecutive *event days*. For example, if the *event days* are 2018-01-01, 2018-01-02, 2018-01-03, 2018-01-12, 2018-01-13, 2018-01-15, then the resulting events are defined by the intervals (start: 2018-01-01, end: 2018-01-03) and (start: 2018-01-12, end: 2018-01-15). We assess the recall of the alert-generation algorithm using the following method: for each (start date, end date) pair, we check if the algorithm produces at least one alert within that interval. In the positive case, the event is considered as *spotted*. The final recall is given by the number of spotted events out of the total number of events. On the other hand, to calculate the precision we iterate over the daily alerts generated by the algorithms. If an alert lies within an event interval defined by a (start date, end date) pair, then it is considered a *hit*; otherwise, it is considered a false alarm. The final precision is obtained as the number of hits out of the overall number of alerts. The F-score is computed with the standard formula: $2 \cdot \frac{precision \cdot recall}{precision + recall}$.



(a) Agglomerative **(b)** kmeans

(c) kmedians **(d)** kmedoids

Figure 14. Precision, recall and F-score achieved by the alert-generation algorithm for different values of *alert threshold* and for each of the four clustering approaches.

We have repeated the experiment for different values of *alert threshold*, in a range between 1 and 5, with higher values producing less alerts and thus making the algorithm more selective. Not surprisingly, recall scores become lower as the threshold is increased, while precision follows the opposite tendency, as shown in Figure 14. The reader notes that we have considered precision, recall and F-score for each of the four clustering algorithms, although the discussion below targets the results obtained with the Agglomerative clustering only. This is a well-known phenomenon in Machine Learning evaluation, commonly referred to as *trade-off between precision and recall*. However, it is remarkable that, with the lowest threshold, our algorithm is able to identify almost all the events listed in the ground truth, while keeping the number of false alarms relatively small (the precision is above 0.5). It is worth noting that, in this specific application field, recall can be considered more important than precision: in fact, for a trader

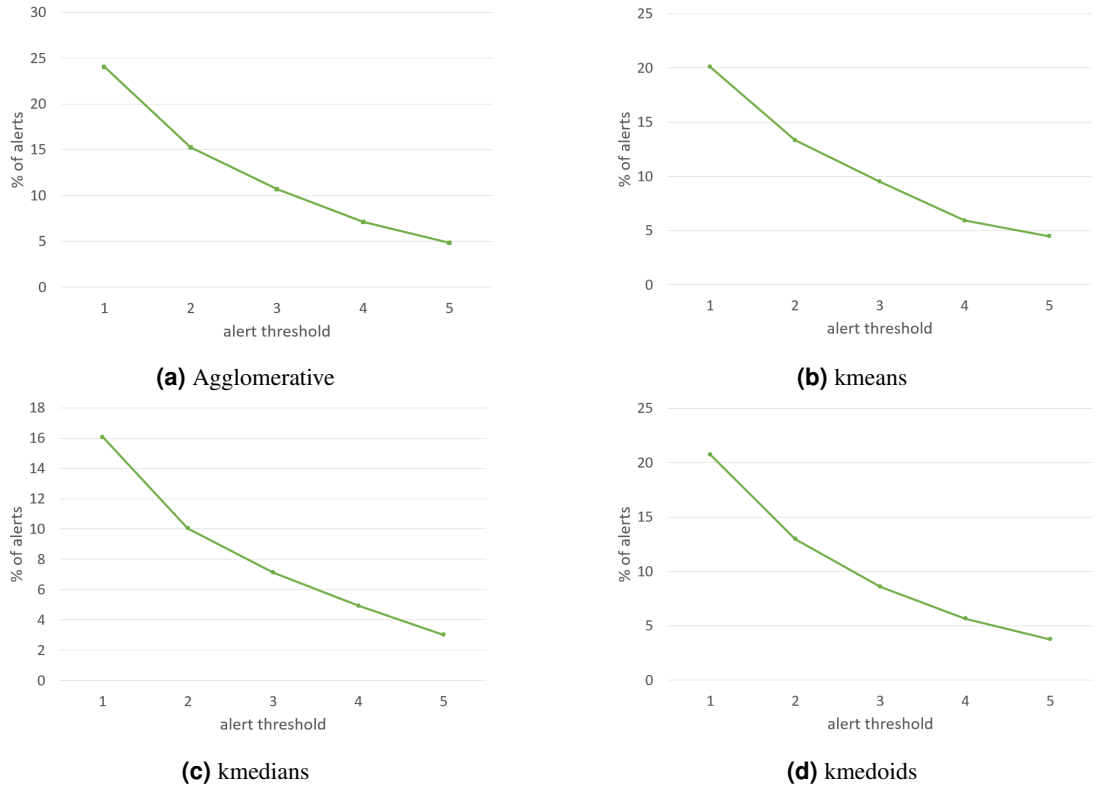


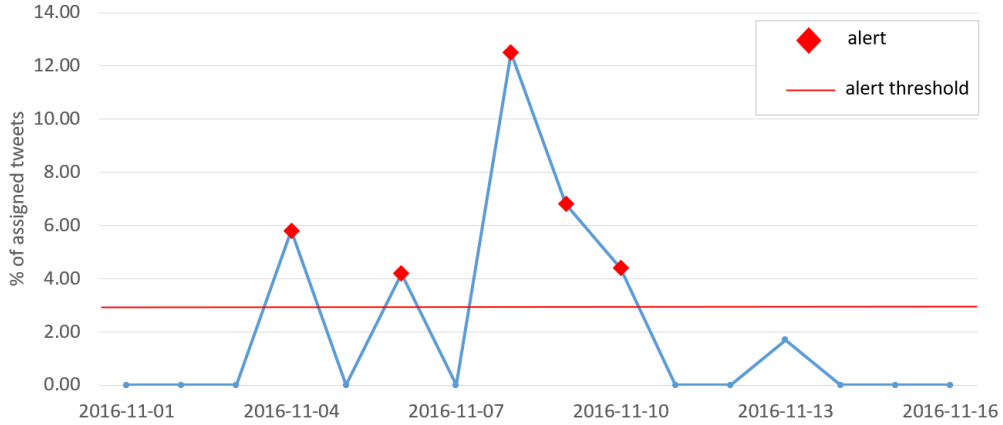
Figure 15. Percentage of alerts produced by the alert-generation algorithm in the time-span considered for the experiments (June 2016 - March 2020), for different values of *alert threshold*.

who relies on the alert-generation algorithm to make informed decisions, receiving some false alarms is arguably a lesser evil than missing relevant warnings about events that actually take place. In order to study further this phenomenon, we manually inspected several clusters that led to a false alarm, in order to understand which kinds of events cause this behavior. In many cases, we observed events like, e.g., quarterly earnings reports, that generate a big "hype" among Stocktwits users, but usually do not produce a proportional impact on the stock price. Furthermore, we calculated the percentage of days marked with an alert out of the whole period on which the algorithm was executed. Figure 15 demonstrates that, for each of the employed clustering algorithms, even with the lower thresholds, the probability of receiving an alert is still reasonably low, confirming that the algorithm is well-aimed.

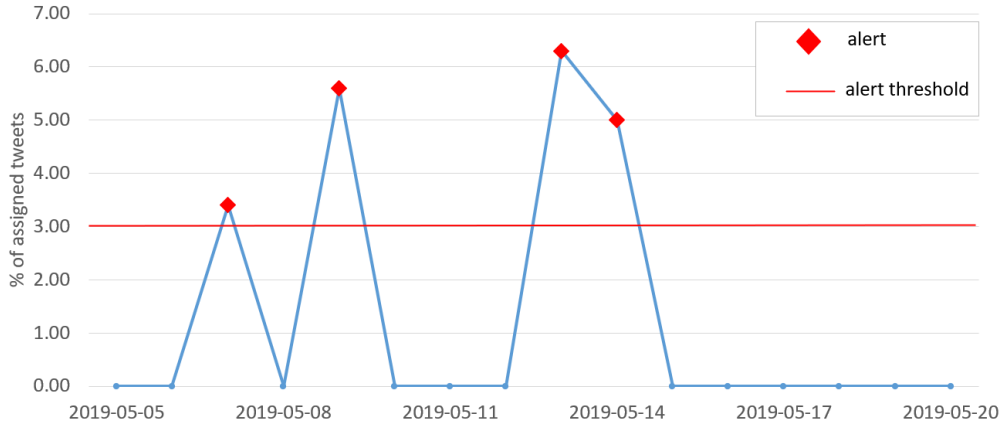
An interesting finding is that, in several cases, the alert is produced with a delay of several days after the actual event took place. This can be partly ascribed to the asynchronism between newswires providers and social media (Petrović et al., 2010; Osborne and Dredze, 2014). In addition to this, in our specific application it is important to take into account the latency between the event itself and its effects on the market. In fact, an event might appear in financial news and on financial-related media only after some time, when its economical consequences manifest themselves. This was the case, for example, for the Covid-19 emergency: American investors, consumers and market in general basically ignored news of the virus outbreak in January 2020, even though several hundred cases had already been reported and Wuhan City (first main site of the virus) had already been quarantined by the Chinese authorities. Nevertheless, starting from February the virus became the main concern of the financial-related press and media and the stock market suffered a violent plunge towards the end of the month due to Covid-19 fears[19].

To conclude the visual inspection of the events discussed in the previous section, Figure 16 illustrates the plot of the percentage of total assigned tweets, whose behaviour determines the generation of the alerts (as explained in Section 3.8). It is straightforward to notice that the curves reach their peaks in correspondence of the date of the event for all three case studies, further confirming the sensitivity of our approach.

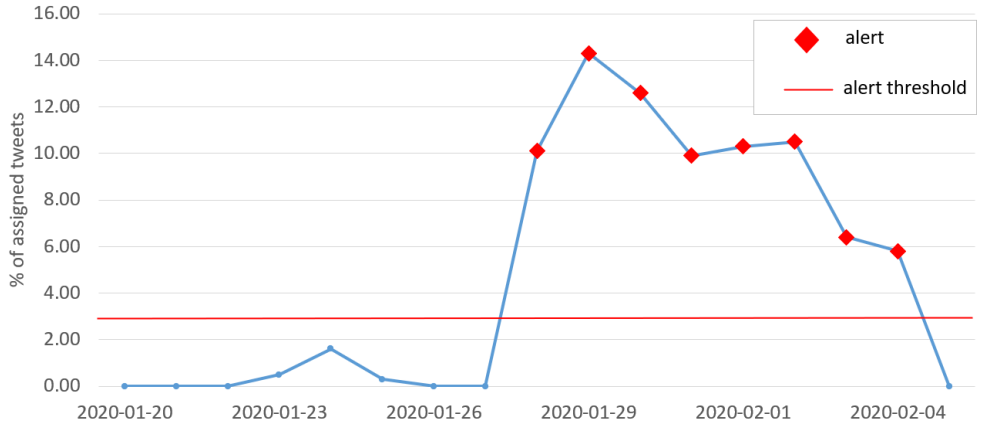
[19] https://www.valuationresearch.com/pure-perspectives/covid-19-event-timeline/



(a) 2016 U.S. Presidential Elections



(b) U.S.-China trade war



(c) Covid-19 outbreak

Figure 16. Plot of the percentage of assigned tweets with respect to the overall number of published tweets, for each day in the interval around the dates of the events included in our qualitative evaluation, respectively. The red markers indicate the generated alerts, while the red horizontal line represents the *alert threshold*.

6 CONCLUSIONS AND FUTURE WORK

In this work, we proposed an event-detection approach tailored for financial applications that leverages the integration of traditional newswires and social media data in order to extract information about real-world events, on a daily basis. Starting from a specialized domain-aware representation of textual data obtained through *ad-hoc* lexicons and word-embeddings, our pipeline is able to identify groups of semantically related news by means of a hierarchical clustering algorithm. An outlier-removal module refines the clusters by discarding misclassified documents, so that a noise-free, meaningful representation of events can be computed. At this point, the news clusters are enriched by data coming from social media, with the goal of estimating the impact of events on public opinions. Despite the defined tweet distance threshold that should avoid this case, it might happen that tweets containing different information are associated with the same cluster triggering the presence of a hot event that would correspond to a false positive. Although this condition has not occurred within our experiments, we will investigate it further in future works we are headed. Finally, by monitoring the activity on social media platforms, an alert-generation algorithm produces warnings to the final users, in order to make them aware of potentially relevant events that have a big resonance on the Internet. The reader ~~notices~~ that to identify the events of a day d our proposed approach generates the lexicon out of news articles and stock data information of previous days up to $d-1$ without looking at the future. This makes our approach suitable for real-time event detection.

One of the advantages of the proposed approach is that, although it is domain-specific, it can be easily extended to various application fields with minimum modifications. However, in this work we described the specific pipeline and experimental framework that we implemented for the financial sphere. More in detail, word2vec models can be trained ad-hoc on text corpora in other languages, as the algorithm itself is not language-dependent. As an example, libraries such as spaCy[20] provide pre-trained word-embedding models in 15 languages. Although the approach is scalable and does not have high computational times, each of its steps can be run on different machines by exploiting its pipeline architecture. Also, big data frameworks such as Apache Spark, Apache Hadoop, ElasticSearch can be leveraged and managed by cloud systems (e.g. Amazon AWS) to further make the approach faster especially if the input dataset grows exponentially.

We validated our approach by means of a qualitative and quantitative evaluation, based on Dow Jones' *Data, News and Analytics* dataset, a stream of data collected from the Stocktwits platform and the stock price time series of the S&P 500 Index. Our experiments show that the approach is effective at identifying clusters of news that correspond to relevant real-world events and at extracting meaningful properties about the associated topic. Besides, the alert-generation algorithm produces warning about *hot* events with a satisfactory accuracy, covering the majority of financial events taking place in the real world and keeping the number of false alarms relatively small. An added value of our evaluation is given by the visual inspection of a selected number of significant real-world events, starting from the Brexit Referendum and reaching until the recent outbreak of the Covid-19 pandemic in early 2020.

One of the applications we envision is the creation of a set of financial indicators that can help improving the accuracy of existing robo-advisory and robo-trading systems. The idea is that identified hot events should be associated to high stock variations and this information might be leveraged to further tune financial forecasting systems.

In the future, we intend to carry out a deeper inspection of the temporal aspects related to the event-detection task. In particular, we want to gain a better understanding of the effect produced by parameters such as the size of the time windows used for the lexicon creation or for the clustering algorithm. Together with this, we will evaluate the timeliness of the alert-generation algorithm, with the goal of reducing the delay of the generated warnings with respect to the actual starting moment of the events. Another aspect that deserves further investigation is the method used to represent social media data in a vector space. Specifically, we intend to refine the representation of tweets by applying pre-processing techniques that are required by the specificity of the language forms commonly employed by users on Internet platforms. These methods include the assessment of the veracity and reliability of the published content and the detection of *slang*, grammatical mistakes, misspellings and abbreviations. Last but not least, we would like to take full advantage of the power and benefit that Semantic Web technologies bring: as such we would like to employ ontologies and best practices of the Semantic Web for the extraction and identification of particular events in order to improve further the obtained clustering. The employment of

[20] https://spacy.io/

big data frameworks previously mentioned should address potential computational or scalability problems we might encounter.

ACKNOWLEDGMENTS

We would like to thank the Centre for Advanced Studies at the Joint Research Centre of the European Commission for guidance and support during the development of this research work.

REFERENCES

Ajao, O., Bhowmik, D., and Zargari, S. (2018). Fake news identification on twitter with hybrid cnn and rnn models. In *Proceedings of the 9th International Conference on Social Media and Society*, pages 226–230.

Allan, J., Carbonell, J. G., Doddington, G., Yamron, J., and Yang, Y. (1998a). Topic detection and tracking pilot study: Final report. In *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, pages 194–218.

Allan, J., Papka, R., and Lavrenko, V. (1998b). On-line new event detection and tracking. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 37–45.

Alvanaki, F., Sebastian, M., Ramamritham, K., and Weikum, G. (2011). Enblogue: emergent topic detection in web 2.0 streams. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, pages 1271–1274.

Aramaki, E., Maskawa, S., and Morita, M. (2011). Twitter catches the flu: Detecting influenza epidemics using Twitter. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1576.

Atefeh, F. and Khreich, W. (2015). A survey of techniques for event detection in Twitter. *Computational Intelligence*, 31(1):133–164.

Beal, M. J., Ghahramani, Z., and Rasmussen, C. E. (2002). The infinite hidden markov model. In *Advances in Neural Information Processing Systems*, pages 577–584.

Becker, H., Naaman, M., and Gravano, L. (2011). Beyond trending topics: Real-world event identification on Twitter. In *Fifth International Conference on Weblogs and Social Media (ICWSM'11)*, pages 438–441.

Boudoukh, J., Feldman, R., Kogan, S., and Richardson, M. (2019). Information, trading, and volatility: Evidence from firm-specific news. *The Review of Financial Studies*, 32(3):992–1033.

Carta, S., Consoli, S., Piras, L., Podda, A., and Reforgiato Recupero, D. (2020). Dynamic Industry-specific Lexicon Generation for Stock Market Forecast. In *Lecture Notes in Computer Science (to appear)*, volume 12565. Springer.

Consoli, S., Darby-Dowman, K., Geleijnse, G., Korst, J., and Pauws, S. (2010). Heuristic approaches for the quartet method of hierarchical clustering. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1428–1443.

Consoli, S., Korst, J., Pauws, S., and Geleijnse, G. (2020). Improved metaheuristics for the quartet method of hierarchical clustering. *Journal of Global Optimization*, 78:241–270.

Daniel, M., Neves, R. F., and Horta, N. (2017). Company event popularity for financial markets using twitter and sentiment analysis. *Expert Systems with Applications*, 71:111–124.

Ding, X., Zhang, Y., Liu, T., and Duan, J. (2015). Deep learning for event-driven stock prediction. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, page 2327–2333.

Dou, W., Wang, X., Ribarsky, W., and Zhou, M. (2012). Event detection in social media data. In *IEEE VisWeek Workshop on Interactive Visual Text Analytics-Task Driven Analytics of Social Media Content*, pages 971–980.

Ein-Dor, L., Gera, A., Toledo-Ronen, O., Halfon, A., Sznajder, B., Dankin, L., Bilu, Y., Katz, Y., and Slonim, N. (2019). Financial event extraction using Wikipedia-based weak supervision. In *Proceedings of the Second Workshop on Economics and Natural Language Processing*, pages 10–15, Hong Kong. Association for Computational Linguistics.

Fan, J. and Lv, J. (2008). Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(5):849–911.

826 Genovese, C. R., Jin, J., Wasserman, L., and Yao, Z. (2012). A comparison of the lasso and marginal
827 regression. *Journal of Machine Learning Research*, 13(68):2107–2143.
828 Gilbert, E. and Karahalios, K. (2010). Widespread worry and the stock market. In *Fourth International*
829 *AAAI Conference on Weblogs and Social Media (ICWSM'10)*, pages 58–65.
830 Hasan, M., Orgun, M. A., and Schwitter, R. (2018). A survey on real-time event detection from the
831 Twitter data stream. *Journal of Information Science*, 44(4):443–463.
832 Heston, S. L. and Sinha, N. R. (2017). News vs. sentiment: Predicting stock returns from news stories.
833 *Financial Analysts Journal*, 73(3):67–83.
834 Hogenboom, A., Hogenboom, F., Frasincar, F., Schouten, K., and Van Der Meer, O. (2013). Semantics-
835 based information extraction for detecting economic events. *Multimedia Tools and Applications*,
836 64(1):27–52.
837 Hogenboom, F., de Winter, M., Frasincar, F., and Kaymak, U. (2015). A news event-driven approach for
838 the historical value at risk method. *Expert Systems with Applications*, 42(10):4667 – 4675.
839 Hu, L., Zhang, B., Hou, L., and Li, J. (2017). Adaptive online event detection in news streams. *Knowledge-*
840 *Based Systems*, 138:105–112.
841 Jacobs, G., Lefever, E., and Hoste, V. (2018). Economic event detection in company-specific news text.
842 In *Proceedings of the First Workshop on Economics and Natural Language Processing*, pages 1–10.
843 Kaiser, G. (2010). *A friendly guide to wavelets*. Springer Science & Business Media.
844 Kaleel, S. B. and Abhari, A. (2015). Cluster-discovery of twitter messages for event detection and
845 trending. *Journal of Computational Science*, 6:47–57.
846 Kaufmann, M. and Kalita, J. (2010). Syntactic normalization of twitter messages. In *International*
847 *Conference on Natural Language Processing*, volume 16.
848 Kumaran, G. and Allan, J. (2004). Text classification and named entities for new event detection. In
849 *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development*
850 *in Information Retrieval*, pages 297–304.
851 Kusner, M., Sun, Y., Kolkin, N., and Weinberger, K. (2015). From word embeddings to document
852 distances. In *32nd International Conference on Machine Learning, ICML 2015*, volume 2, pages
853 957–966.
854 Lam, W., Meng, H., Wong, K., and Yen, J. (2001). Using contextual analysis for news event detection.
855 *International Journal of Intelligent Systems*, 16(4):525–546.
856 Li, Z., Wang, B., Li, M., and Ma, W.-Y. (2005). A probabilistic model for retrospective news event
857 detection. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and*
858 *Development in Information Retrieval*, pages 106–113.
859 Lo, A. W. (2004). The adaptive markets hypothesis. *The Journal of Portfolio Management*, 30(5):15–29.
860 Makrehchi, M., Shah, S., and Liao, W. (2013). Stock prediction using event-based sentiment analysis. In
861 *2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent*
862 *Technologies (IAT)*, volume 1, pages 337–342.
863 Marcus, A., Bernstein, M. S., Badar, O., Karger, D. R., Madden, S., and Miller, R. C. (2011). Twit-
864 Info: Aggregating and visualizing microblogs for event exploration. In *Proceedings of the SIGCHI*
865 *Conference on Human Factors in Computing Systems*, pages 227–236.
866 Marx, V. (2013). The big challenges of Big Data. *Nature*, 498:255–260.
867 Mathioudakis, M. and Koudas, N. (2010). TwitterMonitor: Trend detection over the twitter stream. In
868 *Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data*, pages
869 1155–1158.
870 Mele, I., Bahrainian, S. A., and Crestani, F. (2019). Event mining and timeliness analysis from heteroge-
871 neous news streams. *Information Processing & Management*, 56(3):969–993.
872 Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of
873 words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*,
874 pages 3111–3119.
875 Murtagh, F. (1983). A survey of recent advances in hierarchical clustering algorithms. *The Computer*
876 *Journal*, 26(4):354–359.
877 Musaev, A., Wang, D., and Pu, C. (2014). LITMUS: Landslide detection by integrating multiple sources.
878 In *ISCRAM*, pages 677–686.
879 Nuij, W., Milea, V., Hogenboom, F., Frasincar, F., and Kaymak, U. (2014). An automated framework
880 for incorporating news into stock trading strategies. *IEEE Transactions on Knowledge and Data*

Engineering, 26(4):823–835.

Osborne, M. and Dredze, M. (2014). Facebook, Twitter and Google Plus for breaking news: Is there a winner? In *8th International AAAI Conference on Weblogs and Social Media (ICWSM'14)*, pages 611–614.

Osborne, M., Petrovic, S., McCreadie, R., Macdonald, C., and Ounis, I. (2012). Bieber no more: First story detection using Twitter and Wikipedia. In *Sigir 2012 Workshop on Time-aware Information Access*.

Petkos, G., Papadopoulos, S., and Kompatsiaris, Y. (2012). Social event detection using multimodal clustering and integrating supervisory signals. In *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*, pages 1–8.

Petrović, S., Osborne, M., and Lavrenko, V. (2010). Streaming first story detection with application to twitter. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 181–189.

Petrovic, S., Osborne, M., McCreadie, R., Macdonald, C., and Ounis, I. (2013). Can Twitter replace Newswire for breaking news? In *Seventh International AAAI Conference on Weblogs and Social Media (ICWSM'13)*, pages 713–716.

Ritter, A., Etzioni, O., and Clark, S. (2012). Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1104–1112.

Rokach, L. and Maimon, O. (2005). Clustering methods. In *Data Mining and Knowledge Discovery Handbook*, pages 321–352. Springer US.

Rosa, R. L., Silva, M. J., Silva, D. H., Ayub, M. S., Carrillo, D., Nardelli, P. H., and Rodríguez, D. Z. (2020). Event Detection System based on User Behavior Changes in Online Social Networks: Case of the COVID-19 Pandemic. *IEEE Access*, 8:158806–158825.

Ruiz, E. J., Hristidis, V., Castillo, C., Gionis, A., and Jaimes, A. (2012). Correlating financial time series with micro-blogging activity. In *Proceedings of the 5th ACM International Conference on Web Search and Data Mining*, pages 513–522.

Saeed, Z., Abbasi, R. A., Maqbool, O., Sadaf, A., Razzak, I., Daud, A., Aljohani, N. R., and Xu, G. (2019). What's happening around the world? a survey and framework on event detection techniques on twitter. *Journal of Grid Computing*, 17(2):279–312.

Sakaki, T., Okazaki, M., and Matsuo, Y. (2010). Earthquake shakes Twitter users: real-time event detection by social sensors. In *Proceedings of the 19th International Conference on the World Wide Web*, pages 851–860.

Salton, G., Wong, A., and Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18:613–620.

Schumaker, R. P. and Chen, H. (2009). Textual analysis of stock market prediction using breaking financial news: The AZFin text system. *ACM Transactions on Information Systems (TOIS)*, 27(2):1–19.

Stilo, G. and Velardi, P. (2016). Efficient temporal mining of micro-blog texts and its application to event discovery. *Data Mining and Knowledge Discovery*, 30(2):372–402.

Stokes, N. and Carthy, J. (2001). First story detection using a composite document representation. In *Proceedings of the First International Conference on Human Language Technology Research (HLT)*, volume H01-1030.

Suykens, J. A. and Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300.

Thapen, N., Simmie, D., and Hankin, C. (2016). The early bird catches the term: Combining Twitter and news data for event detection and situational awareness. *Journal of Biomedical Semantics*, 7(1):61.

Tsapeli, F., Bezirgiannidis, N., Tino, P., and Musolesi, M. (2017). Linking twitter events with stock market jitters. *arXiv preprint arXiv:1709.06519*.

van der Maaten, L. and Hinton, G. E. (2008). Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.

Weng, J. and Lee, B.-S. (2011). Event detection in twitter. In *Fifth International Conference on Weblogs and Social Media (ICWSM'11)*, pages 401–408.

Xie, W., Zhu, F., Jiang, J., Lim, E.-P., and Wang, K. (2016). TopicSketch: Real-time bursty topic detection from Twitter. *IEEE Transactions on Knowledge and Data Engineering*, 28(8):2216–2229.

Xing, F. Z., Cambria, E., and Welsch, R. E. (2018). Natural language based financial forecasting: a survey.

Artificial Intelligence Review, 50(1):49–73.

Yand, Y., Pierce, T., and Carbonell, J. (1998). A study on retrospective and on-line event dection. In *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 28–36.

Yang, Y., Zhang, J., Carbonell, J., and Jin, C. (2002). Topic-conditioned novelty detection. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 688–693.

Yates, A., Banko, M., Broadhead, M., Cafarella, M. J., Etzioni, O., and Soderland, S. (2007). TextRunner: Open information extraction on the Web. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pages 25–26.

Zhang, Y., , Jin, R., and Zhou, Z.-H. (2010). Understanding bag-of-words model: A statistical framework. *International Journal of Machine Learning and Cybernetics*, 1(1-4):43–52.

Zhao, Y., Karypis, G., and Fayyad, U. (2005). Hierarchical clustering algorithms for document datasets. *Data Mining and Knowledge Discovery*, 10(2):141–168.

Zhou, D., Chen, L., and He, Y. (2015). An unsupervised framework of exploring events on Twitter: Filtering, extraction and categorization. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI'15)*, pages 2468–2474.

Zhou, P., Cao, Z., Wu, B., Wu, C., and Yu, S. (2018). Edm-jbw: A novel event detection model based on js-id forder and bikmeans with word embedding for news streams. *Journal of Computational Science*, 28:336–342.