


Integrated visual analysis of patterns in time series and text data - Workflow and application to financial data analysis

Information Visualization
2016, Vol. 15(1) 75–90
© The Author(s) 2015
Reprints and permissions:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/1473871615576925
ivi.sagepub.com


Franz Wanner, Wolfgang Jentner, Tobias Schreck,
Andreas Stoffel, Lyubka Sharalieva and Daniel A Keim

Abstract

In this article, we describe a workflow and tool that allows a flexible formation of hypotheses about text features and their combinations, which are significantly connected in time to quantitative phenomena observed in stock data. To support such an analysis, we combine the analysis steps of frequent quantitative and text-oriented data using an existing a priori method. First, based on heuristics, we extract interesting intervals and patterns in large time series data. The visual analysis supports the analyst in exploring parameter combinations and their results. The identified time series patterns are then input for the second analysis step, in which all identified intervals of interest are analyzed for frequent patterns co-occurring with financial news. An a priori method supports the discovery of such sequential temporal patterns. Then, various text features such as the degree of sentence nesting, noun phrase complexity, and the vocabulary richness, are extracted from the news items to obtain meta-patterns. Meta-patterns are defined by a specific combination of text features which significantly differ from the text features of the remaining news data. Our approach combines a portfolio of visualization and analysis techniques, including time, cluster, and sequence visualization and analysis functionality. We provide a case study and an evaluation on financial data where we identify important future work. The workflow could be generalized to other application domains such as data analysis of smart grids, cyber physical systems, or the security of critical infrastructure, where the data consist of a combination of quantitative and textual time series data.

Keywords

Heterogeneous data, time series analysis, frequent financial data analysis, text document analysis, interest point detection, interesting interval patterns, hybrid temporal pattern mining (HTPM), hypothesis generation

Motivation

In many application areas, the key to successful data analysis and the understanding of complex processes is the integrated analysis of heterogeneous data. One example is the financial domain, where time-dependent and highly frequent quantitative data (e.g. trading volume and price information) and textual data (e.g., economical and political news reports) need to be considered jointly in order to find explanations of phenomena. Data analysis tools need to support an

integrated analysis, which allows studying the relationships between textual news documents and quantitative properties of the stock market price series in

Data Analysis and Visualization Group, University of Konstanz,
Konstanz, Germany

Corresponding author:

Franz Wanner, Data Analysis and Visualization Group, University of
Konstanz, Universitätsstr. 10, Konstanz 78457, Germany.
Email: franz.wanner@uni-konstanz.de

increasing amounts of data. For many years, economists tried to figure out how stock markets react to new information. In 1994, Mitchell and Mulherin¹ described a weak relation between the quantity of published news and stock market behavior. A similar approach is taken by Graf,² who uses automatic sentiment extraction of news to show the relationship of sentiment and disagreement with economic variables on a daily basis. In 2011, Bollen et al.³ and Zhang et al.⁴ investigated how Twitter (<https://twitter.com/>) messages may predict the stock market. The authors try to find correlations of the mood reflected in the Twitter data and financial time series developments.

The efficient market hypothesis (EMH) states that all available information is reflected in the prices of financial instruments.⁵ Despite the “body of evidence in support of EMH,”⁵ there are also several counterexamples, for example, the financial crisis in 2007 or other financial market events. These motivated us to have a deeper look into financial news and information possibly hidden in them. The above-mentioned authors try to cover some semantic aspects of a text by filtering out a specific text feature, often the mood or the sentiment, respectively. But is this the only information we can extract and which is worth considering? Recently, Dzielinski⁶ published an article titled “The role of information intermediaries in financial markets.” He describes how companies publish bad news and also the behavior of news agencies in this context. Findings of his work are “that announcements containing bad news are longer and less focused on the originating company than good news” and “companies attempt to ‘package’ bad news and mitigate its negative impact.” Here, “package” refers to the way of writing and the extent and the level of detail of the content. A further result is “that news agencies step in and cut through the packaging by reporting bad company news in a much more concise and focused way.” These statements not only are economically interesting but also serve as a starting point for the development of appropriate analysis techniques. Hence, we are looking for a way to enable an analyst to find and detect text features of interest in conjunction with highly frequent financial time series data of market prices. We believe that such text features may be able to convey part of the *hidden* information. Financial domain experts may use the output of our analysis pipeline as an input for their models for verification purposes. The goal is to find evidence for the observed feature combinations by using economic market models.

Our text analysis approach follows the approach by Oelke:

Most analysis tasks do not require a full text understanding. Instead, one or several semantic aspects of the text

(called quasi-semantic properties) can be identified that are relevant for answering the analysis task. This permits to a targetly [sic] search for combinations of (measurable) text features that are able to approximate the specific semantic aspect. Those approximations are then used to solve the analysis task computationally or to support the analysis of a document (collection) visually.⁷

The pipeline we propose in this article offers the functionality to detect and search for such text feature combinations.

A short description of our two-step workflow is as follows (see Figure 1): in a first step, we search for interesting interval patterns in highly frequent stock data (minute-based). The second step is using these interval patterns to get ordered frequent patterns in combination with news. This process is needed because we are eventually interested in *meta-patterns*. These patterns consist of previously unknown specific text feature combinations. Since the textual news is also contained in the sequential temporal pattern, these features may convey information which affects the stock prices immediately. The analytical challenge is to bring the heterogeneous data together: highly frequent financial time series and text data. Several automated analysis techniques have to be applied, and visualizations are needed to give as much feedback as possible to the user during the analysis process and to enable the analyst to interact with the system. Our design is a combination of visualizations and automated analytical methods.⁸ The heterogeneous nature of our data and the analysis task require multiple views. We follow the design guidelines of Wang Baldonado et al.⁹ and aim at developing a straightforward and interactive system providing transparent analysis, exploration capabilities, and flexibility to compose a complex analysis from various analysis building blocks.

Our research is motivated by the findings of the mostly economic-related research mentioned above. By our tool, we want to give economists the opportunity to identify text properties that affect financial instruments on a minute base. The contribution of this article is to analyze heterogeneous data, that is, identify interesting financial time series intervals and corresponding news features within the analytical task to identify salient text features for hypothesis generation and verification by domain experts. Furthermore, we enable the analyst to gain insight and explore the data patterns interactively in a convenient way.

The remainder of this article is structured as follows: section “Related work” gives a brief overview of the two-step pipeline. In section “Conception of the proposed two-step analysis workflow”, we describe the first step: the detection of interesting time series interval patterns. In section “Visual-interactive analysis to

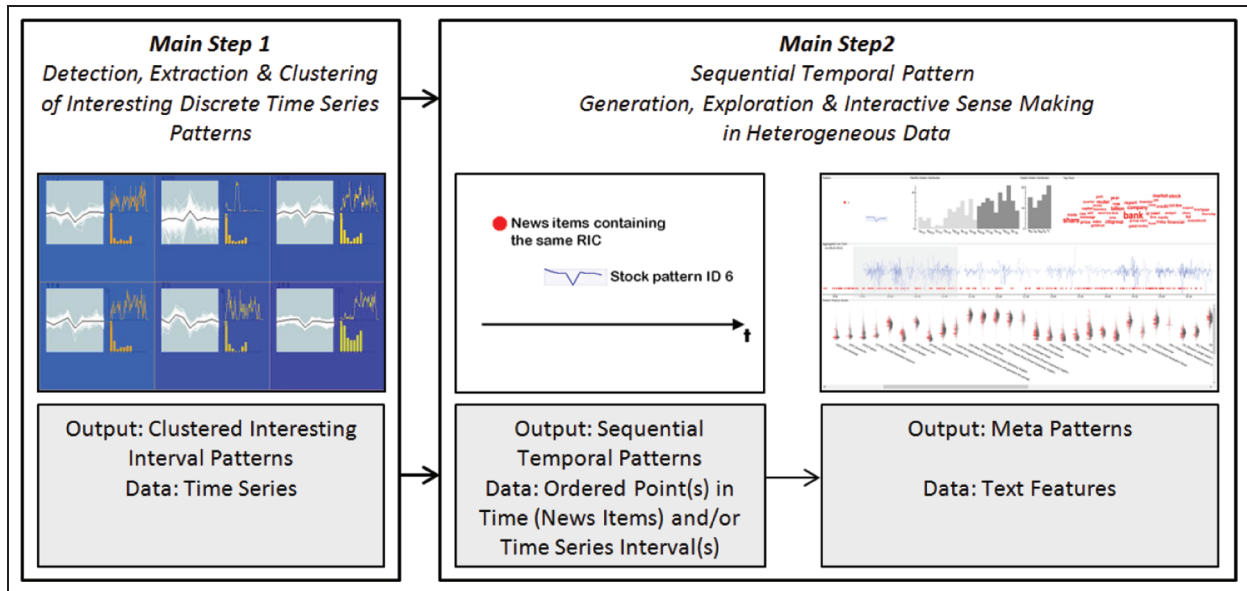


Figure 1. Proposed main two-step workflow. First, interesting quantitative local time series patterns are detected in a visual-interactive approach (step 1). Then, in a second workflow, news data are correlated using an a priori method with the time series patterns to get sequential temporal patterns (left portion, step 2). The components of a sequential pattern are as follows: the news is represented by a red item annotated with n which is followed in the example by the interval pattern 6. Within this step, text features are extracted from textual news data referring to a company. A detail-on-demand exploration and navigation interface enables the analyst to interpret the sequential pattern and find meta-patterns (right portion, step 2). Meta-patterns are formed through the red highlighted distribution representation which shows text features of news belonging to the sequential pattern. The feature distributions represented in gray belong to news referring the same company but are not part of the sequential pattern. The workflow design emphasizes the possibility to use each main step as a stand-alone application. The visual representation within each step facilitates this.

determine quantitative time series interval patterns”, we define the second analysis step. We search for sequential temporal patterns also containing news and support the user in the sense-making process of finding interesting text features. This may serve as a starting point for further research. The usefulness and applicability is shown in section “Sequential temporal pattern analysis and meta-pattern exploration”. In section “Conclusions and future work”, we conclude this article and give an outlook on future work.

Related work

Our pipeline components are related to a number of previous approaches in visual analysis of time series and textual data. Additional related works will be discussed where appropriate throughout the technical sections.

Visual analysis of patterns in time series

Many approaches proposed to date focus on the analysis of time series data using visual methods. An excellent overview of visualization techniques for time series is presented in Aigner et al.¹⁰ Exploration of large time

series can rely on interactive or automatic approaches or combinations of both. An interactive exploration system for time series data is TimeSearcher¹¹ which allows querying a repository of time series for user-specified patterns of interest. The query is done by the analyst through so-called timeboxes, that is, rectangular frames which can be flexibly modified. In general, all well-known interaction techniques can support navigation and interactive exploration of time series data.¹² Automatic analysis of time series data often involves data reduction, for example, cluster analysis or interest point or interval detection. The self-organizing map (SOM) algorithm¹³ is a well-known visual cluster method which can reduce large datasets, with a wealth of visualization possibilities.¹⁴ The SOM method has been successfully deployed in many applications, including financial data analysis.¹⁵ It can also be effectively used in semi-supervised, interactive analysis tasks.¹⁶ Recently, several approaches address the analysis of local patterns in time series, that is, search for interest points that denote some particular intervals of interest. For example, Kincaid¹⁷ studied a visual overview of interest points detected in long time series, and Schreck et al.¹⁸ proposed a pixel-oriented

approach for analyzing interest points in time series at different scales.

Sequential temporal pattern generation and exploration

A priori algorithms are implemented by toolkits such as Weka¹⁹ or Rapidminer (<http://rapid-i.com/content/view/181/>). Approaches to visually search for text feature rules are presented in Wong et al.,^{20,21} where the components of the rules were all extracted from the same domain, that is, from a homogeneous input dataset. We look for association rules which are defined over heterogeneous data. Here, the building blocks for the a priori rule extraction are in turn patterns extracted from heterogeneous data, that is, time series and news document streams. In our approach, we use the HTPM algorithm²² to detect combined point- and interval-based patterns for further interactive exploration in order to extract meaningful text feature combinations for hypotheses generation. The authors of the HTPM show the applicability of their algorithm in a real case study of the financial data and news domain. They focus on split and dividend announcements, and their interval event patterns are statically predefined. In the case study, their goal is to show the prediction power of such news contained in the hybrid patterns. A similar approach can be found in Fan et al.,²³ where the original HTPM algorithm is extended to take the duration of an event into account and discuss several methods to do so. In our application, the duration of an event can be limited interactively by the user to get more meaningful patterns.

News feature extraction

Park et al.²⁴ try to isolate the different aspects of news automatically to get more insights into news and show the different viewpoints reflected in them. We also aim at determining different text features of news. The text features are extracted using the *sxTransformer framework*²⁵ which was implemented for readability analysis of text documents. We are able to calculate about 130 text features on different structural levels and domains such as quantitative and non-quantitative linguistic features, multiple information retrieval-related features, and various other miscellaneous features. A commercial tool with even more text analysis functionality is TextQuest[™].²⁶ One of its core capabilities is also readability analysis.

Financial text analysis systems

Various approaches can be found in Nikfarjam et al.²⁷ They present a survey on text analysis approaches in

the financial domain. Included is also the AZFin text system of Schumaker and Chen.²⁸ They apply machine learning to predict stock market returns 20 min after a news item was published. The AZFin textual analysis covers the content of the news using three different representations: bag of words, named entities, and proper nouns. The system design aims on identifying “the important article terms”²⁸ by using these word-based text representations. According to their results, proper nouns performed best in all three stock prediction evaluation scenarios. Most existing approaches combine steps as text content analysis and processing of financial time series for labeling purposes in order to train a classification model.^{27,29} An improved machine learning approach is shown in Li et al.²⁹ They include also recent history prices in their model. A good overview of sentiment analysis applied to financial markets can be found in Schumaker et al.³⁰ In section “Visual-interactive analysis to determine quantitative time series interval patterns”, we discuss the qualitative differences between AZFin and our application in more detail.

Conception of the proposed two-step analysis workflow

In this section, we describe the basic idea and functionality of our two-step workflow. The subsequent section “Visual-interactive analysis to determine quantitative time series interval patterns” will detail the first step, and in section “Sequential temporal pattern analysis and meta-pattern exploration”, we explain the second step.

The workflow is an encompassing analysis pipeline aiming at relating patterns found in quantitative time series and text-oriented data. Effectively, each of the two main steps of the integrated analysis workflow is a pattern analysis workflow in itself. Within the first workflow, *quantitative time series interval patterns* are identified by means of interest point detection and clustering. Within the second workflow, the quantitative interval patterns of the first workflow and the occurrence of news are correlated by an a priori analysis providing the most relevant sequential co-occurrences of interval patterns and news (*sequential temporal pattern generation*). We extract *time-dependent text features* from *all* news items referring to a particular company and visualize the distribution of text features in a density plot and a matrix view. Text features which belong to news contained in the sequential pattern are shown as the red distribution in the density plot and pixel representations in the matrix view. The visual-interactive approach identifies *meta-patterns* that can then be interpreted by the analyst to form



Figure 2. Workflow for pattern detection in time series data. Input is a set of time series. A visual-interactive analysis process aims at finding local interval patterns of interest from the time series. These, in turn, are then input to the second step, namely, correlating it with textual news (see section “Sequential temporal pattern analysis and meta-pattern exploration”).

hypotheses about dependencies and correlations. The integrated analysis is enabled by visual exploration techniques, including detail-on-demand inspection of time series and textual properties. Figure 1 illustrates the basic workflow.

The *design goals* of our devised workflow are to provide a modular and transparent usage of analysis and visualization methods during the analysis process. Another goal is to provide interaction in order to modify search parameters and explore the results in a convenient manner.

Visual-interactive analysis to determine quantitative time series interval patterns

The first stage of our overall analysis workflow aims at finding a set of interesting local patterns from an input set of time series. We define a pattern in this context as a sub-sequence in a time series which is both *interesting* (in the sense of a statistical interest point detector) and occurs *frequently* in the set of time series. The analysis aims at reducing a large set of time series to a smaller set of interpretable chunks of information. This is done by a combined automatic and interactive analysis which includes appropriately defined visual representations and interactions. Figure 2 illustrates the workflow based on detection, clustering, and selection steps as detailed in the following.

Interest point detection

First, we detect a set of local interest points in time series data as the basic element for subsequent analysis. A local interest point needs to be characterized by some criterion of what constitutes interestingness. We follow our concept from Schreck et al.,¹⁸ which is based on applying a variant of the Bollinger Band detector and similar approaches.³¹ The idea is to find local points in the time series which exhibit some sort of outlying behavior. In the Bollinger Band techniques, this is implemented by comparing each value in a time series with a moving average of the values in a defined neighborhood around that point. At every point where the current value is higher or lower than the moving average by some margin, an interest point is reported.

Figure 3 (top left) shows an example of the Bollinger Band technique: Whenever a given time series value (the gray line chart) exceeds the moving average (shown by the yellow line chart) by the user-given threshold, an interest point is reported (shown by red bars). We support the identification of interest points by visualizing detected interest points in response to interactively setting parameters across a small multiple view of all time series in the dataset. As we typically look not at a single but many time series, we use a small multiple approach to present a comparative view on interest points detected in the line charts of many time series (see Figure 3, top right). Therefore, the user can compare the interest points across time series in response to different parameter settings for threshold and moving average window size.

For analysis of interest points among many time series and alternative detection parameter settings, we also give a pixel-oriented view on the data. The view is constructed by showing all time series in an array of pixel rows, where each line represents a time series, and each set pixel represents a detected interest point for a given parameter choice. This allows to efficiently compare the occurrence of interest points across time series. Yet, there remains the problem of finding appropriate parameter settings for stable interest point detection. To this end, we nest the pixel-oriented interest point view by another small multiple representation, where the detection results are shown for different parameters of the detection. Figure 3 (bottom) shows six pixel displays of the interest points detected in a set of time series. The left (right) column shows results for low (high) threshold, and the bottom (top) row shows results for low (high) moving window size. From this comparative display, it is possible to effectively find stable parameter ranges (e.g. the circled patterns in Figure 3 (bottom right) identify that for higher detection thresholds, interest points are detected for many time series in approximately the same time regions).

Interest point clustering

The output of the interest point detection is a set of points. The set of interest points may be very large

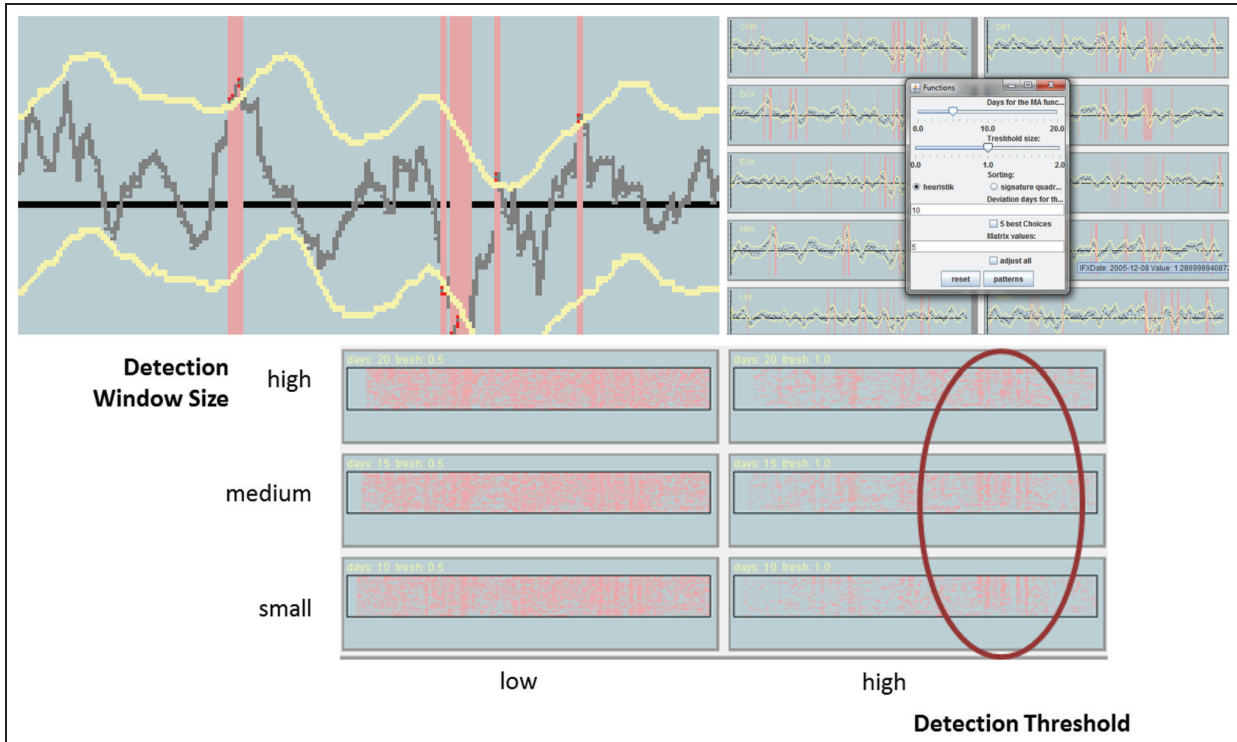


Figure 3. Visual-interactive detection of local interest points in time series. First, local interest points are detected using a Bollinger Band detector (top left). Here, the x-axis shows the time on a minute basis. The gray line shows the return of the stock. Whenever the return exceeds the Bollinger Band, the chart is highlighted in red. Then, detected interest points are visualized in context of several time series using line charts and markers for interactively selected detection parameters in a small multiple comparative display (top right). For large time series, a pixel-based representation supports scalability and can be used as a comparative view for the detection of appropriate settings in the parameter space (bottom). Circled are stable interest points for different window sizes (for all rows) and high thresholds (for the right column). Specifically, it appears that all time series show occurrence of interest points at approximately the same time position.

and includes noise in the local time series. Therefore, we reduce the set of interest points by identifying local time series intervals around the detected points. To this end, we apply the well-known SOM¹³ algorithm to cluster and visually organize the set of detected interest points. For each detected interest point, we extract a small time series interval centered on the respective interest point. Typically, we select nine values per interval (interest point itself ± 4 data values), but this size depends on the resolution of the data. The obtained set of local time series segments is now clustered by the SOM algorithm for the visual cluster analysis. Technically, we normalize each time series segment linearly to span the $[0, 1]$ interval and consider the result as the input vector to train the SOM. We visualize the output as a two-dimensional (2D) map of time series clusters as computed by the SOM. Specifically, we show the time series prototypes and

cluster member time series' by an overlay in a grid-based view. The set of clusters is, by properties of the SOM method, sorted for similarity in the layout (see Figure 4, left).

The SOM result is a first step toward obtaining a smaller set of meaningful time series interval patterns. However, meaningfulness of local interest patterns can also stem from temporal correlations between the patterns. As an example, a pair of clusters, which co-occur frequently across time or with a fixed temporal lag, can be considered more interesting than other patterns which are not correlated. To this end, we support color-coding of interval patterns to the small multiple of the time series view. Figure 4 (middle) shows an example where the user has identified two interval patterns (denoted in red and blue) which co-occur frequently together across the time series (see circled parts).

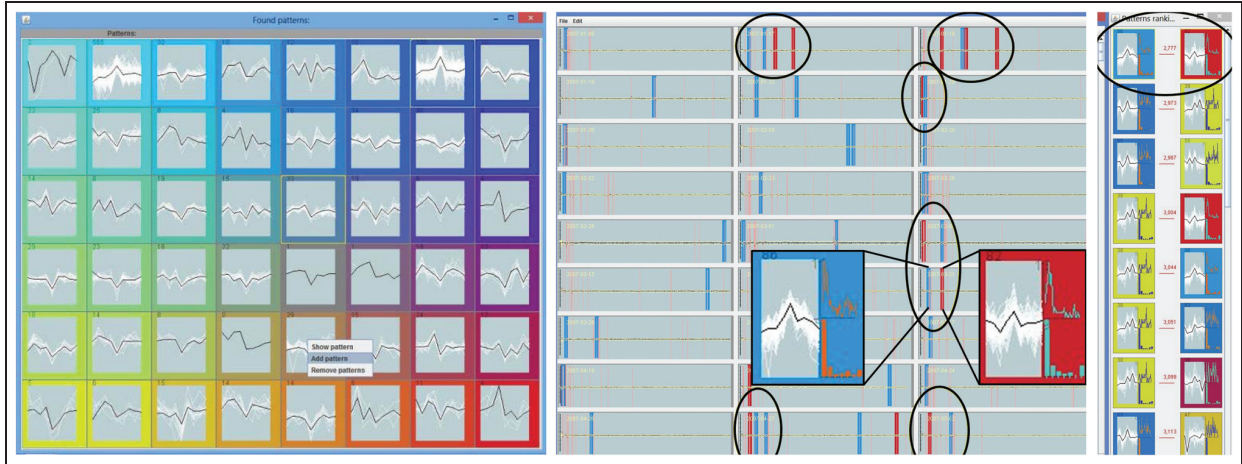


Figure 4. Analysis for meaningful groups of local interest points based on visual cluster analysis of local time series intervals around identified points. The self-organizing map (SOM) method is employed for visual cluster analysis (left). A 2D color-coding approach allows to link selected SOM patterns of interest with their occurring pattern sequences in a time chart view (middle). In this view, selected SOM clusters are represented as color markers which indicate both the position on the SOM and their relative distance measured on the SOM grid. Also, users can perceive pattern details by hovering over the marker view, giving the frequency of the respective patterns across the time axis and per time series. Besides manual selection of important patterns, we also propose an automatic scoring function which suggests a ranking of patterns for exploration (right).

Proposed pattern scoring heuristic

By means of the approaches described above, users can manually search for local time series interval patterns based on local interestingness measures (in the sense of the Bollinger detector) and also frequency and correlation measures (based on cluster analysis and color-coding). In addition, we support fully automatic filtering for relevant local interval patterns based on a heuristic search. Specifically, we define a compound selection score D defined on pairs of SOM clusters (c_1, c_2) as follows

$$D(c_1, c_2) = dt(c_1, c_2) \times \frac{1}{dc(c_1, c_2)} \times \frac{1}{ds(c_1, c_2)} \quad (1)$$

The score consists of three terms aggregated in a product sum. $dt(c_1, c_2)$ denotes the average normalized temporal distance between the patterns across all time series. The smaller this value, the closer the patterns are co-occurring, indicating a more interesting relationship between them. $dc(c_1, c_2)$ denotes the distance between the cluster positions in the SOM grid. The more distant they are in the SOM grid, the more dissimilar they are to each other. We assume that more dissimilar pattern pairs are more interesting, in particular, if they occur close to each other in time, which indicates a different local behaviors. Finally, $ds(c_1, c_2)$ is an optional term that encodes the semantic distance

between the corresponding overall time series, if available. The term is used to encode background knowledge. For example, the semantic distance between two time series of stock prices of companies can be measured by the sectoral similarity of the markets the companies are operating in. We consider a pair of interval patterns more interesting, if the patterns are semantically more dissimilar. Note that we set this term to 1.0, in case no such background information is available. In effect, the smaller the score $D(c_1, c_2)$, the more interesting the pair of clusters is considered by our heuristic.

We use this score to produce a ranking of cluster pairs across all time series in the database. We visualize the ranking of clusters as local time series interval pattern glyphs. The glyphs are sorted by interestingness and include the frequency of the pattern across the time axis and per time series (see Figure 4, insets in the middle portion). The latter views can be used in addition to assess properties of the local interval patterns across the time axis and across time series. Figure 4 (right) illustrates a ranking of patterns obtained with our heuristic score. Note that the patterns occur from rather distant areas of the SOM (linked by color-coding in blue and red, see background color map in Figure 4 (left)) and also occur close in time (see Figure 4 (middle) for the proximity of patterns).

Resulting time series patterns

The result of the preceding steps is a reasonably small number of local interval time series patterns which have been identified in a semi-automatic way by the user. The patterns consist of the shape of a time series interval. They are interesting according to various criteria which can be inspected and controlled by the user, who is assisted by automatic search and interactive parameter steering as well as the dynamic response of all involved visualization components. The most interesting clusters (time series interval patterns) are then exported as input to the second analysis step in our combined workflow.

Sequential temporal pattern analysis and meta-pattern exploration

Our second analysis module generates frequent sequential temporal patterns in time series by using previously detected interval (see section “Visual-interactive analysis to determine quantitative time series interval patterns”) and news events. A news item contains a timestamp and a reference to one or several companies. We use the HTPM approach²² to preserve the ordering of the pattern components. In addition, we store the temporal occurrences of the patterns. Each pattern can be analyzed interactively by the user, enabling the analyst to determine the discriminating features (meta-pattern). The goal is to provide an opportunity for generating hypotheses of dependencies, influences, and root causes for further verification and evaluation by the analysts. The interactivity is needed to learn how the patterns are connected in time and which features form meta-patterns.

In contrast to the AZFin text system²⁸ and many other applications published so far, our multi-feature application allows flexible time spans between a news item’s publication date and a certain interesting stock pattern. Furthermore, it is envisioned that stock patterns are followed by a news release. We also allow the reverse case where a news item may be preceded by an interesting stock pattern. Beyond this, more sophisticated sequential patterns consisting of several news and interesting interval patterns are possible as long as they are above the support threshold. Different from the three news representations of AZFin and other approaches (e.g. sentiment-based analysis), we extract about 130 text features for each news item. The goal is to find feature-sets which are significantly frequent within a sequential temporal pattern. Since a stock pattern is interesting according to the applied interestingness measures of main step 1 (see Figure 2), these found distinctive feature-sets (meta-patterns) can be used in further analyses to better understand the

influence of news on certain stocks in terms of writing style, grammar, and content.

Data

The data are heterogeneous. On one hand, we have news data as time events with high-dimensional text features. On the other hand, we have stock data which are interval based and have a high temporal resolution (minute by minute).

Stock data. For the stock data, we have gathered records from January 2007 to May 2009 for 29 stocks (New York Stock Exchange; <https://nyse.nyx.com/>) which are identified by their Reuters Instrument Code (RIC; <http://www.reuters.com/>). Only trading days are available, mostly from 9:30 a.m. to 4:00 p.m. The data are recorded per second, but for our analysis, we aggregate the records to the minute level. The HTPM algorithm does not require any special resolution in time, but we want to find fast occurring reactions in the market. This is in contrast to most related work which focuses on correlations or patterns on a daily basis. Each minute includes a starting price (p_{start}) and an ending price (p_{end}). We calculate the return (R) value as our main parameter of interest

$$R = \frac{p_{end} - p_{start}}{p_{start}} \quad (2)$$

News data. The news is provided by Reuters and is available in the time period from June 2007 to December 2010. The news was published in English, and each news item is linked to one or more RICs. This linkage is done manually by the authors of the articles. In total, we have about 210,000 news articles and extract about 130 text features for each news item (see section “Related work”).

Pipeline

To be flexible and efficient in generating hypotheses, our pipeline is divided into five steps (Figure 5). Although the input step is mainly intended for preprocessing and aggregating the input data, the preprocessing step itself is designed to carry the data into a data structure the HTPM algorithm can work with. In the processing step, the HTPM algorithm finds sequential temporal patterns. In the analysis step, the user is able to select patterns he or she is interested in, which can be exploited in the following step. The users may explore the patterns to find clusters or outliers of text features which belong to the sequential pattern.

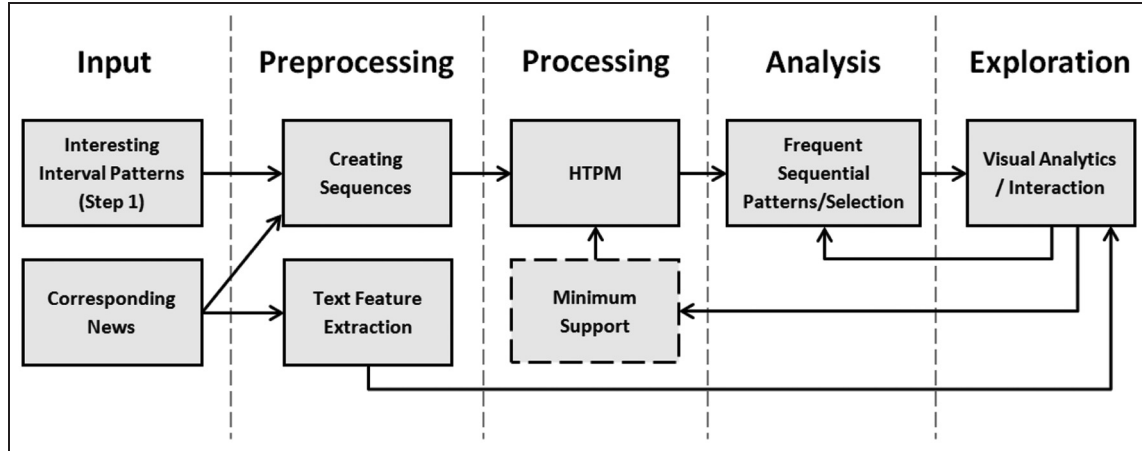


Figure 5. The sequential temporal pattern generation and exploration pipeline.

Input. The input consists of two event categories, namely, interval patterns (see section “Visual-interactive analysis to determine quantitative time series interval patterns”) and news events. While interval patterns can be of arbitrary length, news items occur at specific points in time. The ID of the stock events is determined by the cluster ID which is generated by the SOM (Figure 6(a)). The stock events are grouped by similarity, so the time series have similar shapes. For our analysis, the interval events have a length of 9 min which creates nine data values. The length is defined during the interesting interval detection, see section “Interest point clustering.” For each day, we load the corresponding news in terms of date and RIC.

Preprocessing. Each trading day is represented as a sequence. The HTPM algorithm calculates a support value for each pattern. The news items as well as the interval events are transformed into events with a specific occurrence within a sequence. For the interval events, we store the return values for each point in time and add it as metadata to the event. The metadata of the news consist of text features which become important in the exploration step. We extract about 130 features on different structural levels using the *sxTransformer framework*.²⁵ Besides basic features such as the part-of-speech category or word stems, we are also able to extract linguistic units and several quantitative linguistic features. Furthermore, we can analyze the grammatical structure, the readability, and the vocabulary richness. Information retrieval-related features and different noun-to-verb ratio features complete the picture. We delete sequences that do not contain any news items since we are mainly interested in identifying news–interval patterns.

Processing. The HTPM algorithm is capable of finding sequential patterns that are above a given minimum support. Similar to other a priori methods, it bases on the “anti-monotone property.”²² First, the algorithm searches for one-event interval-based and point-based patterns. The support (s) is defined as the number of sequences (S) where the pattern (p) occurs, divided by the total number of sequences (equation (3)). If a pattern occurs several times in a sequence, it will be ignored. Then, two-event patterns are joined. Only patterns which are above the given minimum support are retained. Afterward, the iterative process joins the patterns consisting of the same prefixes. This means that they have an equal ordering when the last event occurrence is deleted. This happens as long as no new combined sequential temporal pattern has a support value above the given minimum support

$$s = \frac{|\{p|p \in S\}|}{|S|} \quad (3)$$

Analysis. The output of the HTPM algorithm consists of all sequential patterns. To give the user an intuitive understanding of the relations,²² these patterns are visualized as illustrated in Figure 6(b). Given the case that nesting appears, we can extend the representation by the one the HTPM authors use for such patterns.

Exploration. By selecting a sequential temporal pattern, the user is able to explore the pattern in detail. All occurrences of the pattern are visualized in an aggregated line chart. The user may filter the data in terms of temporal duration of the sequential pattern. Further information about exploration and interaction capabilities is described in section “Visual analysis tool”.

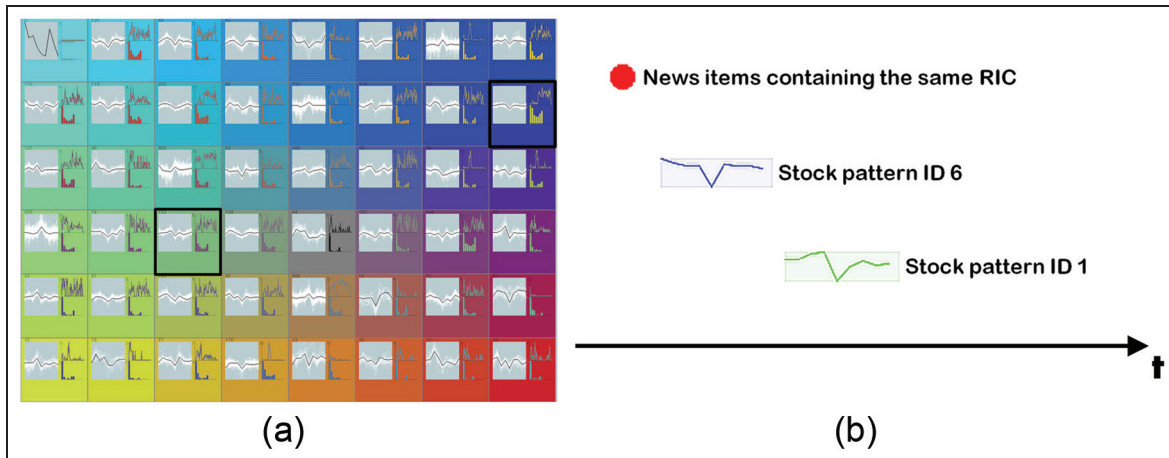


Figure 6. The input data of main step 1 and a resulting sequential temporal pattern of main step 2—(a) output of main step 1: clustered interesting interval patterns in the self-organizing map. This is the input for main step 2. The black framed patterns are an example of how they could occur in (b), and (b) is an schematic example of a sequential temporal pattern consisting of three events without nesting. ID 6 and ID 1 are interval-based patterns representing a stock pattern. The news items represented by a red dot are a point-based event.

Visual analysis tool

To give the user more analysis capabilities and to enable exploring and following first clues, we implemented different data views and filters in an interactive system (Figure 7). The sequential temporal pattern area (1) gives the user a first intuitive feedback on the overall chronological pattern sequence. It is represented sequentially from left to right through its components. In addition, a prototype of the interesting interval pattern is shown. Whenever the user changes a filter, a second prototype with a lower opacity will be shown. News is represented in red. The color of the stock pattern is determined by the SOM visualization. The occurrence per day of the week and the monthly pattern distribution (2) enables the user to observe when a pattern occurs in time. In addition, months or weeks can be selected or deselected, so the data can be explored in the particular time span the analyst is interested in. All these actions result in a direct feedback and will change the other visualizations. The word-cloud³² (3) shows the most frequently occurring words in the news (except stop words), which provides a semantic feedback to the analyst. In order to explore the content of the news, the user is able to drag a flexible sliding window (light gray, close to (4)) over the daily sequential pattern distribution. The word-cloud is updated according to the news contained in the window. Together with the monthly and day-of-week filter possibilities, the user can examine news groups down to the minute level. The word-cloud provides a semantic understanding of the content. It is possible to remove words out of the word-cloud and add them to

a stop word list. Furthermore, by hovering over the words, the news will be highlighted in the aggregated line chart (4). By clicking on the words, the underlying news will be opened in order to read the full content. Another possibility to get the content of the news is to click on the news representation in the aggregated line chart. The aggregated line chart shows all selected news and interval patterns. While the interval patterns are represented as line charts, the news events are drawn as small triangles on the bottom of the chart. The text feature view (5) shows the text features from the news on a certain RIC as a sort of a density plot,³³ an alternative representation of a boxplot and modified for our purposes. The plots are sorted according to their distinctiveness which is measured by one of the offered three different statistical tests. A standard unpaired t -test, the Wilcoxon–Mann–Whitney test, and the Kolmogorov–Smirnov test. While the t -test and the Wilcoxon–Mann–Whitney test measure the distinctiveness of the average, the Kolmogorov–Smirnov test measures how significant two distributions are differing. In our case, the two samples are the news which are part of the pattern (red) and the news that belong to the RIC but are not part of the pattern (gray). The test is performed for each dimension, and the features are then ranked according to their p value in an ascending order. The user may also provide an α -value which will remove the features with a p value greater than the α -value. Currently, we are calculating about 130 text features generated by the *sxTransformer framework*, and the user is able to select or add features. By exploring the visualization, the user is able to

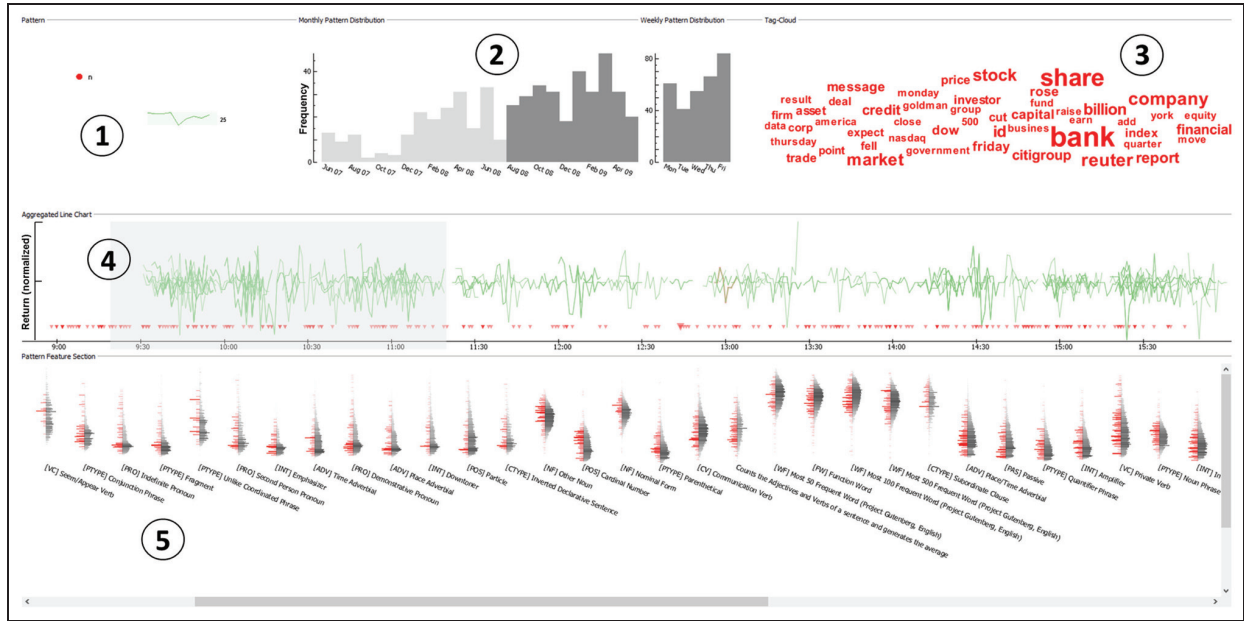


Figure 7. The visual analysis tool for the exploration of sequential and meta-patterns—(1) sequential temporal pattern area, (2) distribution in time (month or days of week), (3) word-cloud, (4) aggregated line chart (here you can see one pattern highlighted in red and its associated news), and (5) density plots of the text features showing the distribution of the news within the pattern (red) and the outside of the pattern (gray). The normalized values of each feature are mapped on the length of the bars and the opacity. This makes it easy to find diverse distributions. In addition, the statistical tests sort the density plots according to their significance. The overview shows the found sequential pattern, consisting of a news followed by the interesting interval pattern 25 (upper left) of Citigroup, a large US bank. ID 25 of the pattern is determined by the SOM. This pattern has a support of 67% (see equation (3)), meaning that this pattern occurs at 67% of all analyzed days.

find text features or feature combinations (*meta-patterns*) which are representative for a specific sequential pattern. The user has multiple opportunities to interact with the visual analysis tool (see Figure 8). At first, he or she is able to filter data according to the months or day of weeks using the histograms. Furthermore, limitations in terms of duration are possible. The duration is the full length of the pattern. Optionally, the user may also use only the shortest pattern for each sequence which in our case is equal to trading days. Another filter is to search for specific key words in the news. A simple fuzzy string matching on lemmatized words is applied here. Several options can be used to make the visualization more appealing. Even though the colors are determined by the SOM visualization and the news is red by default, they can be changed later. Also, the thickness and the opacity of the lines in the aggregated line chart can be modified. The choice of the statistical test and the α value influence only the density plots and pixel visualizations.

In a second window, the user may evaluate further patterns within the features and for each news which belongs to the pattern. We try to support this with a pixel visualization (Figure 8).

The features are represented per column, while each news item is represented in a row. The features are ordered according to the statistical test which is also used for the parallel coordinates. Each feature value in each news item is represented as a square. The opacity of the square shows the feature value where a full saturation means a high value and no saturation a low value. The values are normalized for each feature over all the news in the pattern. The news can be ranked by their importance or by their published date. The importance for each news is measured with the p value (p) of a standard t -test. Since a small p value is desired, we subtract the p value from 1 when the test is performed with the news that is not in the pattern ($n \notin p$). The two p values are summed and then weighted with the p value ($1 - p_{feature}$) of the feature. The test to calculate this p value can be selected by the user. This gives distinctive features more weight in the ranking score of the news. The news is then sorted in a descending order according to their scores (see equation (4))

$$\sum_{\text{over all features}} (1 - p_{n \notin P} + p_{n \in P}) \times (1 - p_{feature}) \quad (4)$$

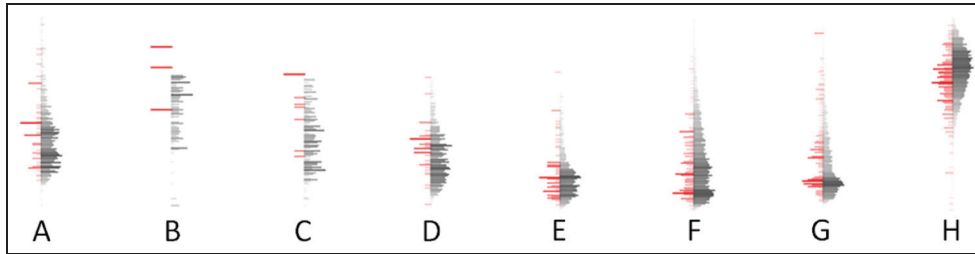


Figure 9. A meta-pattern consisting of eight discriminating text features—A: unlike coordinated phrase, B: direct question, C: inverted yes or no question, D: seem or appear verb, E: conjunction phrase, F: particle, G: pronoun, and H: function word. No filters are applied when we are taking the screenshots. Only the duration of the sequential pattern is limited to 60 min.

conjunction phrases (E) stay low. In addition, *Particles* (F) and *pronouns* (G) seem to be mostly equally used within the selected pattern. However, their distributions show differences. One exception is made for the last feature *function words* (H). Here, the distributions show a shift. Such shifts can also be determined by other statistical tests such as the *t*-test or the Wilcoxon–Mann–Whitney test since these are testing for differences in the average of the distributions. Function words are a combination of different part-of-speech tagged tokens and a wordlist.³⁵ This list consists of different lists: list of *do* forms, of *have* forms, of *be* forms, and a list of *modals*. All lists contain the contracted and negated forms. Function words are known as “grammatical words” which represent “a grammatical or structural relationship with other words in a sentence” (<http://grammar.about.com/od/fh/g/functionword.htm>, accessed 4 August 2013).

Evaluation

In order to test the expressiveness of the text features, we provide a prediction of how likely news belongs to patterns based on their textual features. This likelihood is calculated with a diverging score based on positive and negative probabilities of the patterns. According to our hypothesis, if a news item belongs to a company but is not close enough to an interesting interval pattern, its features can only be weak signals. The closer a news item is to an interesting interval pattern, the higher the expected significance. The probabilities of news are calculated based on two kernel density estimations of the feature domain for positive and negative news. Positive news is news that mentions the company and belongs to the pattern, whereas negative news also belongs to the company but is not detected as a part of the sequential temporal pattern in our analysis. The density estimation is based on a Gaussian kernel and the selected bandwidth according to Silverman’s “rule of thumb”.³⁷ The difference

between the positive and negative density estimations is then used as an indicator for the particular temporal pattern.

Assuming that the stock returns incorporate the knowledge of investors, we tested the features on their effect on stock returns. Based on a simple economic linear regression model, we calculated for each feature, its significance on the stock return and discarded all features with a *p* value > 1%.

We expected features strongly indicating a sequential temporal pattern to explain parts of the observed stock patterns. Unfortunately, in Figure 10, we observe different results and textual features indicating that different text styles or language do not correlate with stock patterns according to our experiments. We have done this experiment for six stocks and the resulting 17 patterns (news followed by a stock pattern and vice versa), and there was neither a common subset of features for a single company nor for a single sequential pattern and, therefore, inevitably not for one industry or all together.

One explanation could be the linear regression. We take the features and measure the significance on the stock return (e.g. Figure 10 and Figure 11). It might improve the results testing on the entire stock pattern and not only the stock return. Then, the patterns could be grouped together according to their appearance.

Another explanation of these results could be that the writing style of financial news is less important than its content. Further experiments have to be conducted measuring the amount of new content to verify this hypothesis.

Conclusion and future work

In this article, we propose an integrated pattern detection workflow to explore heterogeneous data for hypotheses generation. We bring together quantitative time series and text feature data to give analysts a new perspective on relevant data. Compared to the case

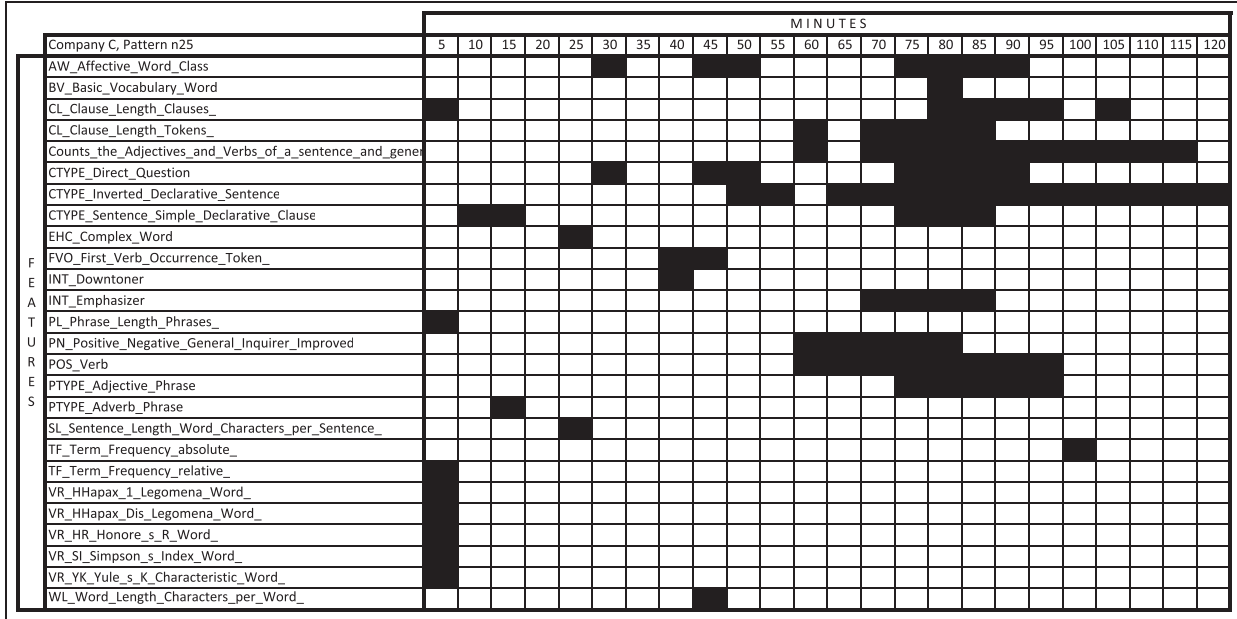


Figure 10. These are the resulting features of the linear regression model for the sequential pattern of a news item followed by the stock pattern 25 used in sections “Sequential temporal pattern analysis and meta-pattern exploration” and “Use case.” Compared to Figure 9, which shows the features for news 60 min before the stock pattern occurs, there is almost no common subset.

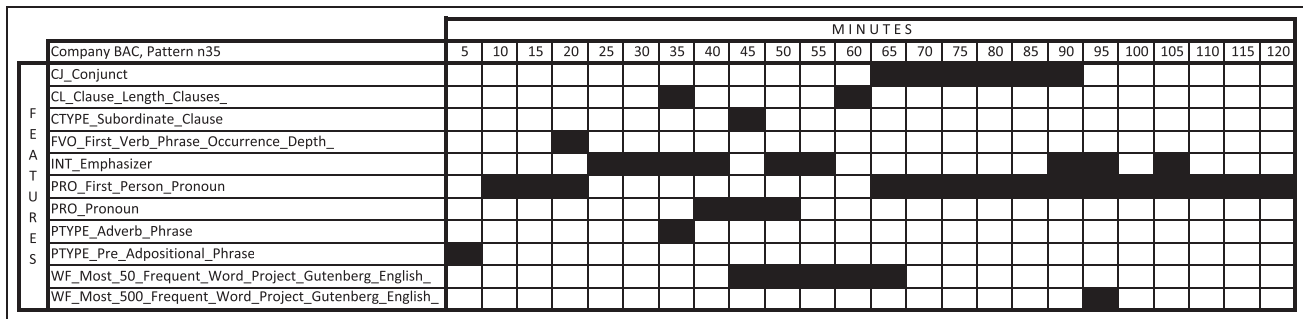


Figure 11. These are the resulting features of the linear regression model for the sequential pattern of a news item followed by the stock pattern 35 of the Bank of America Corporation. There are almost no coinciding features with Figure 10.

study showed in Wu and Chen,²² we use no statically predefined interval patterns. Beyond this, the more in-depth text analysis with visual representation and interactive exploration expands the real case study they present.

The fact that we are able to detect interesting news features within the use case shows the usefulness of our application from an analytical point of view. According to the applied evaluation in section “Use case,” the correlation between patterns detected in the news text space and in the stock time series space is indicative but as of yet not statistically significant. Our

system allows the flexible definition of analysis parameters and, by its interactive nature, provides chances to find starting points in both news and time series spaces for further detailed analysis. The general problem of automatically relating news and stock patterns is a difficult one, as it involves many parameters such as specification of lead or lag time intervals, thresholds, and a multitude of candidate features. Furthermore, financial markets are highly dynamic systems which evolve over time, and it may be questionable if any dependency found will remain stable over time. We aim to conduct more evaluation in the future,

including experts from the financial analysis domain, which could help to further improve the system.

Our approach can also be used in other domains where relationships are presumed and the relevant factors are still unknown. It can provide new insights and serve as a starting point for hypotheses generation purposes. We are currently applying it in the field of energy supply to find relationships between Twitter posts, weather, and power supply system conditions and the electricity output, which highlights the generality and effectiveness of our proposed pipeline.

In the future, this application may lead to new research in the particular domains. An expert user study is planned to get more insight into the analysts needs. To extend the analysis, we integrate different industries' comparison views and a market overview. We want to implement different algorithms for the detection of interesting time series intervals and we plan to apply them to different economic time series (e.g. trading volume).

Acknowledgement

We would like to thank Ferdinand Graf for his initial advice on the extended parts of this publication.

Note

This is an invited paper with extended contents from the previous publication Relating interesting quantitative time series patterns with text events and text features presented at the Conference on Visualization and Data Analysis (VDA) 2014, San Francisco.

Funding

This work has partly been funded by the Research Initiative “Visual Analytics of Text Data in Business Applications” at the University of Konstanz.

References

- Mitchell ML and Mulherin JH. The impact of public information on the stock market. *J Financ* 1994; 49(3): 923–950.
- Graf F. *Mechanically extracted company signals and their impact on stock and credit markets*. Technical report, Department of Economics, University of Konstanz, 2011.
- Bollen J, Mao H and Zeng X. Twitter mood predicts the stock market. *J Comput Sci* 2011; 2(1): 1–8.
- Zhang X, Fuehres H and Gloor PA. Predicting stock market indicators through Twitter “I hope it is not as bad as I fear”. *Proced: Soc Behav Sci* 2011; 26: 55–62.
- Investopedia. *Efficient Market Hypothesis—EMH*, http://www.investopedia.com/terms/e/efficientmarket_hypothesis.asp (2013, accessed 17 July 2013).
- Dzielinski M. The role of information intermediaries in financial markets, <http://ssrn.com/abstract=2266173>
- Oelke D. *Visual document analysis: towards a semantic analysis of large document collections*. PhD Thesis, 2010, <http://nbn-resolving.de/urn:nbn:de:bsz:352-opus-123373>.
- Keim DA, Kohlhammer J, Ellis G, et al. *Mastering the information age-solving problems with visual analytics*. Goslar: Eurographics 2010.
- Wang Baldonado MQ, Woodruff A and Kuchinsky A. Guidelines for using multiple views in information visualization. In: *Proceedings of the working conference on advanced visual interfaces*, Palermo, 24–26 May 2000, pp. 110–119. New York: ACM.
- Aigner W, Miksch S, Schumann H, et al. *Visualization of time-oriented data* (Human-computer interaction series). London: Springer, 2011.
- Hochheiser H and Shneiderman B. Interactive exploration of time series data. In: Jantke KP and Shinohara A (eds) *Discovery science*. Berlin, Heidelberg: Springer, 2001, pp. 441–446.
- Ward MO, Grinstein GG and Keim DA. *Interactive data visualization: foundations, techniques, and applications*. Natick, MA: AK Peters, 2010.
- Kohonen T. Essentials of the self-organizing map. *Neural Network* 37: 52–65.2013.
- Vesanto J. SOM-based data visualization methods. *Intell Data Anal* 1999; 3(2): 111–126.
- Deboeck G and Kohonen T (eds). *Visual explorations in finance: with self-organizing maps*. Berlin, Heidelberg: Springer, 1998.
- Schreck T, Bernard J, Tekuov T, et al. Visual cluster analysis of trajectory data with interactive Kohonen maps. *Palgrave Macmillan Information Visualization* 2009; 8: 14–29.
- Kincaid R. Signallens: Focus + Context applied to electronic time series. *IEEE T Vis Comput Gr* 2010; 16(6): 900–907.
- Schreck T, Sharaliev L, Wanner F, et al. Visual exploration of local Interest points in sets of time series. In: *Proceedings of the IEEE symposium on visual analytics science and technology (Poster Paper)*, Seattle, WA, 14–19 October 2012, pp. 239–240. New York: IEEE.
- Hall M, Frank E, Holmes G, et al. The WEKA data mining software: an update. *ACM SIGKDD Explor Newsl* 2009; 11(1): 10–18.
- Wong PC, Whitney P and Thomas J. Visualizing association rules for text mining. In: *Proceedings of the 1999 IEEE symposium on information visualization (InfoVis '99)*, San Francisco, CA, 24–29 October 1999, pp. 120–123, 152. Washington, DC: IEEE Computer Society, 1999.
- Wong PC, Cowley W, Foote H, et al. Visualizing sequential patterns for text mining. In: *Proceedings of the IEEE symposium on information visualization, 2000 (InfoVis*

- 2000), Salk Lake City, UT, 9–10 October 2000, pp. 105–111. New York: IEEE.
22. Wu S-Y and Chen Y-L. Discovering hybrid temporal patterns from sequences consisting of point- and interval-based events. *Data Knowl Eng* 2009; 68(11): 1309–1330.
 23. Fan SX, Yeh J-S and Lin Y-L. Hybrid temporal pattern mining with time grain on stock index. In: *Proceedings of the 2011 fifth international conference on genetic and evolutionary computing (ICGEC)*, Xiamen, China, 29 August–1 September 2011, pp. 212–215. New York: IEEE.
 24. Park S, Lee S and Song J. Aspect-level news browsing: understanding news events from multiple viewpoints. In: *Proceedings of the 15th international conference on intelligent user interfaces*, Hong Kong, 7–10 February 2010, pp. 41–50. New York: ACM.
 25. Oelke D, Spretke D, Stoffel A, et al. Visual readability analysis: how to make your writings easier to read. *IEEE T Vis Comput Gr* 2012; 18(5): 662–674.
 26. Social Science Consulting. TextQuest—software, <http://www.textquest.de/pages/en/general-information.php?lang=EN> (2013, accessed 17 July 2013).
 27. Nikfarjam A, Emadzadeh E and Muthaiyah S. Text mining approaches for stock market prediction. In: *Proceedings of the 2nd international conference on computer and automation engineering (ICCAE)*, Singapore, 26–28 February 2010, vol. 4, pp. 256–260. New York: IEEE.
 28. Schumaker RP and Chen H. Textual analysis of stock market prediction using breaking financial news: the AZFin text system. *ACM T Inform Syst* 2009; 27(12): 1–12.
 29. Li X, Wang C, Dong J, et al. Improving stock market prediction by integrating both market news and stock prices. In: *Proceedings of the database and expert systems applications*, Toulouse, 29 August–2 September 2011, pp. 279–293. Springer.
 30. Schumaker RP, Zhang Y, Huang C-N, et al. Evaluating sentiment in financial news articles. *Decis Support Syst* 2012; 53(3): 458–464.
 31. Von Landesberger T, Bremm S, Schreck T, et al. Feature-based Automatic Identification of Interesting Data Segments in Group Movement Data. *Information Visualization* 2014; 13(3):190–212.
 32. Viégas FB and Wattenberg M. Timelines tag clouds and the case for vernacular visualization. *interactions* 2008; 15(4): 49–52.
 33. Kampstra P. Beanplot: a boxplot alternative for visual comparison of distributions. *J Stat Softw (Code Snippets)* 2008; 28: 1–9.
 34. Quirk R, Greenbaum S, Leech GN, et al. *A grammar of contemporary English*. Oxford: Oxford University Press, 1972.
 35. Biber D. *Variation across speech and writing*. Cambridge: Cambridge University Press, 1988.
 36. About.com. Grammar & Composition, <http://grammar.about.com/> (2013, accessed 22 July 2013).
 37. Silverman BW. *Density estimation for statistics and data analysis*, vol. 26. Boca Raton, FL: CRC press, 1986.