A Semantic Web Approach for Visualization-Based News Analytics

Maarten Jongmans, Viorel Milea, and Flavius Frasincar

Econometric Institute, Erasmus School of Economics, Erasmus University Rotterdam
P.O. Box 1738, 3000 DR Rotterdam, the Netherlands
289471mj@student.eur.nl, milea@ese.eur.nl, frasincar@ese.eur.nl

Abstract. In order to understand news, dependency patterns between objects in (economic) news items have to be detected. We propose a framework which makes it possible to discover these patterns, and support the observations with statistical analysis. Based on these patterns, alerts can be generated based on emerging news. These alerts can then be used to manage (equity) portfolios. We test our framework based on historical data. The tests show statistically significant results supporting the idea that it is possible to discover such dependency patterns between objects in news items.

1 Introduction

News emerge continuously from a variety of sources and geographic locations. Publishers such as Reuters provide huge amounts of news in digital formats. For knowledge workers it is getting increasingly important to get news fast and in an easy to use manner. News items can have a reasonable value to entities such as banks. Being the first to know the newest (economical/world) news gives you a competitive edge over the competition. In this way, for example, a bank can respond immediately to (expected) changes in equity prices.

Technologies for structuring plain-text, usually on the Web, are grouped under the Semantic Web umbrella [11]. The technologies used to present semantics related to data are the Resource Description Framework (RDF) [20] and the Web Ontology Language (OWL) [10]. These languages provide data in such a way that it is interchangeable and machine understandable. RDF is a general-purpose language for representing information on the Web. OWL is an ontology language that describes the meaning of concepts and is more expressive than RDF. When data is properly described using one of these languages machines can interpret the data and reason with it.

Annotating news is only increasing the amount of data available. In order to get a good overview of the large amount of available news and their semantic content one needs to provide selection procedures that focus at one visualization aspect at-a-time. Most of the examples which are publicly available contain only one layer of data: news items. The current approaches to news visualization are either geographical by creating a view on a map or make use of a timeline

to plot the temporal dimension. A service like Google Finance [19] combines the presentation of stock prices and news items. This is still too limited in our opinion especially because semantic information captured inside news items is not used.

The goal of this paper is to develop a framework that can be used to create visualizations of annotated news items. This framework provides the user with a timeline which can display several different objects at one time and can be used to calculate the dependency between those objects. The primary goal of the framework is to let the user observe the correlation between the objects in the different layers of the timeline. We use data which consists of news items about the companies in the FTSE 100 / 250 [3], made available by Reuters, over the period January 2007 until June 2007. These news items are processed and annotated by ViewerPro. ViewerPro is a tool built by Semlab [7], which is a company specialized in processing news items.

The outline of the paper is as follows. In Sect. 2 we present work related to the goal of this paper. In Sect. 3 we introduce the framework we proposed for visualization-based news analytics. In Sect. 4 we evaluate the introduced framework by means of a tool that we have developed for this purpose. Last, we conclude in Sect. 5.

2 Related Work

Visualization of news can be done in many different ways. Using different technologies, dimensions or data sources, several solutions have been proposed. A data source can be a news repository, a map, or just a financial site with equity prices. Some solutions are able to present relations between the data sources and their data types, like the relation between news and equity prices. A data type can be a news item, an equity price, or a technical indicator. In this chapter we provide an overview of these solutions.

A recurring subject is how to combine different data sources and data types in a comprehensive overview. This can be useful, for example, for management wanting to navigate easily through data without losing context information. Two of the data types we want to use for our framework are 'stock prices' and 'news'. The idea of the relation between stock prices and news items is analyzed in [13]. The same paper also analyses the correlation between news and stock prices, especially when the news occurring is negative, there is a bigger drift in equity price then when there is positive news.

An ontology is a formal representation of a set of concepts within a domain and the relationships between those concepts. In the world of the Semantic Web numerous methods for visualizing ontologies are proposed [18]. Some of these methods are interesting for our research, because these methods can be used for data selection purposes. All of these methods rely on a 2D view to display an ontology. One application that has recently been developed, using such an 2D view, is Hermes [16, 17]. Hermes uses the Prefuse library [6] to display a graph

used for building SPARQL queries [23]. Figure 1 shows the visualization of the ontology items.

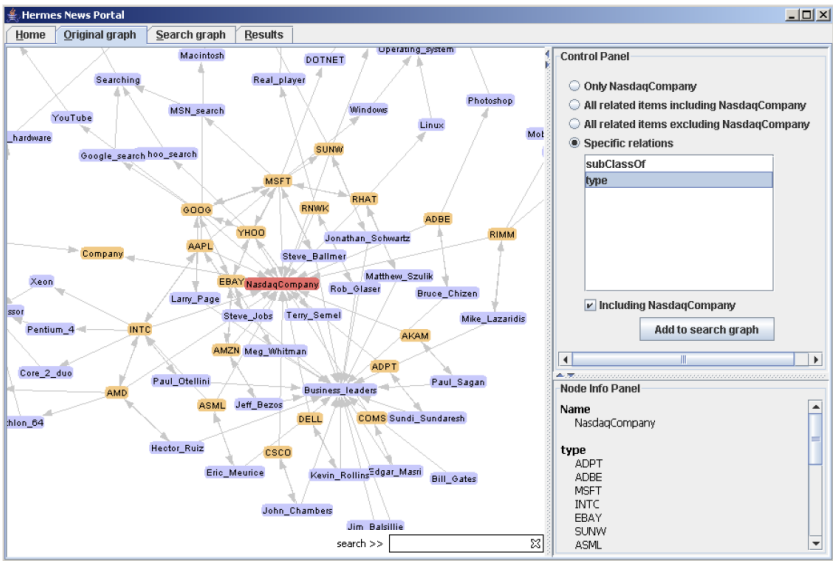

Fig. 1. Visualization of ontology items in Hermes

Another tool which can be used to view ontologies is Protégé. Protégé is an editor that supports graphical views of semantic data. With a plugin like OntoViz [24] you can show the data as a simple graph. IsaViz [22] is a visual environment for browsing and authoring RDF models, represented as directed graphs and can be compared to a tool as Protege with plugins enabled.

Web-related visualization tools are also available. The THOR framework [14] is such an example where a temporal dimension is used. THOR displays news items on a timeline similar like we propose for our framework. The timeline used in THOR is a framework developed by the SIMILE project [9].

BreakingStory is an interactive system visualizing changing news [15]. The news in this framework can be filtered and results can be shown over time. BreakingStory also provides a geographical filter and is able to find related stories across geographically diverse news sources. BreakingStory is using a modified version of the Lucene search engine to look for terms and sentences [5].

TextMap [8] is a portal that tracks references to people, places and things appearing in news items and analyzes meaningful relationships between them. The tasks are done using the Lydia system. Lydia creates a relation model between persons, places, and things by using natural language processing (NLP). To support the NLP method Lydia also uses several statistical methods to analyze entity frequencies and co-locations [21].

Our approach differs from the other solutions as it allows the display of events and other objects, not only news, in a temporal dimension. Moreover our solution can track dependency patterns between the displayed objects.

3 Framework

In order to detect patterns between events and other objects in news items we plan to use visualization techniques. To create such visualizations we developed a framework, called VRBO, in which one can view different types of objects in different ways. VRBO means 'Visualizing Relations Between Objects' and supports five different types of objects. The different types of objects are briefly described below.

3.1 Objects

The visualized objects have data types which can be used as input for timeline layers.

News items All of the news items are related to the economic domain.

Events An event is based on a pattern of annotations. Suppose a news item contains a text like "Tesco is going down by 5%". Then the system can create an event called 'stock price decrease' based on the pattern [*company*, *term*, *percentage*].

Stock prices Each company has associated a stock price at a certain moment in time (end of day is used here).

Stock price changes These changes are calculated based on the price object. This is done by taking the closing price of the current day minus the closing price of the day before divided by the closing price of the day before.

Technical indicators An object can be a technical indicator. A technical indicator is a rule that gives a (trade) signal at a certain moment in time.

3.2 The Timeline

To be able to analyze news through time our framework introduces news browsing. The timeline provided by the SIMILE project [9] offers good basic functionality for this purpose.

This timeline makes it possible to have different bands with different data (objects) with a different sorting method. Different bands with different sorting methods mean that each horizontal layer in the timeline can include data sorted by day, week, month, or year. An example of the timeline is given in Fig. 2. In this figure one can observe a timeline with three layers each having a different (background) color as it represents a different type of data: news items and stock prices. Layer A contains the news sorted by day, later B shows the change in the stock price sorted by day and layer C contains the news sorted by month. Creating layers as C makes it possible for a user to scroll through time without

much difficulty. This is done by synchronizing layers A and B with layer C. When you scroll layer C, layers A and B are automatically adjusted.

Fig. 2. The SIMILE timeline

3.3 Pattern Detection

The primary goal of the framework is to let the user observe the correlation between the objects in the different layers of the timeline. In order to support the visual observations we implemented several statistical methods.

Correlation. The correlation supports the users visual observations by calculating if there exists a correlation between the occurrence of different objects. We calculate the correlation using the following formula:

$$r = \frac{1}{n-1} \sum_{i=1}^{n} \frac{x_i - \overline{x}}{s_x} \frac{y_i - \overline{y}}{s_y} \tag{1}$$

For every observation in the first data set the value x_i is subtracted with the average of that column ($\overline{x}$) divided by the standard deviation (s_x). This is then multiplied by the same computation for the other data set (y). All these calculated values are then summed up and multiplied by $\frac{1}{n-1}$, where n is the total number of observations available.

Hypothesis Testing. To test if the stock price changes after the occurrence of an event (or an other object), we suggest two different hypothesis testing methods. To test these methods hypothesis need to be formulated.

Two Related Samples Test. For this test the following hypothesis can be formulated:

H_0: mean closing price after the event is the same as mean closing price before the event

H_A: mean closing price after the event > mean closing price before the event

The two related samples test is used when phenomena (like prices) are measured twice. In our case before and after an event. The (closing) price can be compared between the day an event happens and the closing price the day before. To perform this comparison we can use a method in which the difference is found between each matched pair of observations. The formula calculating the t-value is:

$$t = \frac{\overline{D}}{\frac{S_D}{\sqrt{n}}} \tag{2}$$

where $\overline{D} = \frac{\sum D}{n}$ and $S_D = \sqrt{\frac{\sum D^2 - \frac{(\sum D)^2}{n}}{n-1}}$. In this formula n is the number of events, D is the difference between the price before and after an event and D^2 is the squared difference of the price before and after an event.

McNemar test. For this test the following hypothesis can be created:

H_0: probability price increases after event is the same as the probability of price decrease after event (P(A) = P(D))

H_A: probability price increases after event is the higher as the probability of price decrease after event (P(A) > P(D))

This test is especially useful with the measurement before and after the same subject [12]. In this case, we measure the price before and after an event. The significance is tested by using a fourfold table of frequencies to represent the first and second set of measurements:

After calculating the χ^2 this value needs to be compared with a critical value. The critical value is based on α and the degrees of freedom [2]. We suggest an α of 0.025 for a one tailed test, while degrees of freedom (df) is 1 for this test. We check if the calculated value is higher than the critical value. If this is true H_0 is rejected and the conclusion is that there is a higher probability of a price increase than of a price decrease after an event. The difference between this test and the two related samples tests is that this test does not assume a normal distribution of the variables.

4 Evaluation

To evaluate the framework we propose we have built a tool. This tool is based on the framework and has most of the proposed methods implemented. In this

evaluation we discuss the configuration options and show how to use the framework.

The tool is created using the Java programming language. Using the Jena [1] framework for accessing the RDF repository, the Apache Tomcat server is serving servlets containing search pages and results. Other languages used primarily to support the input of data are HTML, CSS, and Javascript. The javascript library jQuery [4] is used to traverse HTML documents, handle events, and perform animations supporting our query selection process.

The data used for the evaluation of the tool contains news items about Tesco, which is a United Kingdom-based international grocery and general merchandising retail chain. During the evaluation the tool deals with more than 1GB of data, which is six months of analyzed and annotated news data. This data includes news, annotations, and events. The data is provided by Semlab, who analyzed the (annotated) news items with their software ViewerPro [7]. The news data set is about companies listed on the FTSE 100/250 [3].

4.1 Test Setup

To test the framework we do two hypothesis tests and a correlation test. The user starts each test setting the scope and timeframe (step 1 & 2). A select box contains all the companies available in the data set. After selecting the scope a timeframe is selected. Last, the tests are added to the experiment. Based on this, configuration a timeline is created containing news items, events and significant stock price changes. Significant changes in stock prices are all changes higher than a specific level (in this case changes higher than 1%).

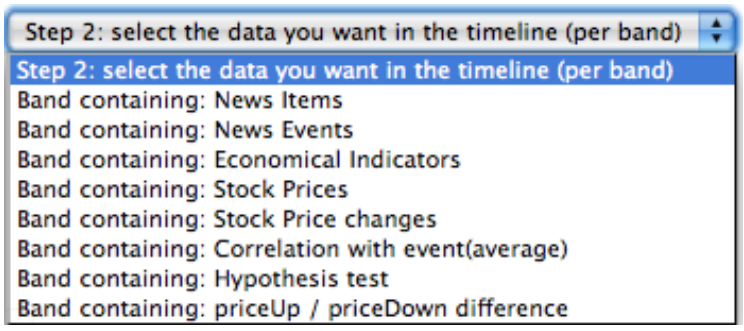

Fig. 3. Selecting an hypothesis test

4.2 Tests

We have done several tests with different test methods. For all the tests the timeframe is set to January 1st 2007 until February 28 2007. The relation between

the event 'companySharesUp' and the stock price is used as our illustrative example. Our own visual observations are that when this event occurs the stock price is often increasing.

Correlation Test. The tests make use of a threshold such that all prices changing more than 1.0% are assigned as 'significant change', all prices changing less than 1.0% do not have a 'significant change'. The result of the correlation test is 0.31. This means that correlation exists, but that is is not high. Perfect positive correlation exists with a correlation of 1.

Two Related Samples t-test. For this test the following hypothesis can be defined:

H_0: mean closing price after the event is the same as mean closing price before the event

H_A: mean closing price after the event > mean closing price before the event

The result of this test is 2.08 with 11 degrees of freedom (d.f.). The critical value with an α of 0.05 and d.f. of 11 is 1.795885. As $2.08 > 1.795885$, H_0 is rejected. This means that the mean closing price is on average higher after an event then the day before the event.

χ^2 McNemar Test. For this test the following hypothesis can be defined:

H_0: probability price increases after event is the same as the probability of price decrease after event (P(A) = P(D))

H_A: probability price increases after event is the higher as the probability of price decrease after event (P(A) > P(D))

The result of the McNemar test is 4. With a critical value of 3.84146 where $\alpha = 0.05$ and df = 1, the hypothesis H_0 is rejected ($4 > 3.84146$). These tests confirm our own visual observations that the stock prices are increasing after the 'companySharesUp' event.

4.3 Experiment Results

With the result of the tests we have performed we can make rules which can then be used in analyzing new news items. Rules take the following form: 'if event A happens event B happens'. Based on the tests described in section 4.2 we can define the following rule (based on the following formal notation: $A, (A \rightarrow B)$):

$CompanySharesUp, (CompanySharesUp \rightarrow$ increasing equity price).

This means that when the event 'CompanySharesUp' happens the equity price is increasing. Another rule which we created using our tool is:

$CompanyJointVenture, (CompanyJointVenture \rightarrow$ increasing equity price).

We observe, as in Fig. 4, that when this event happens, the stock price is increasing.

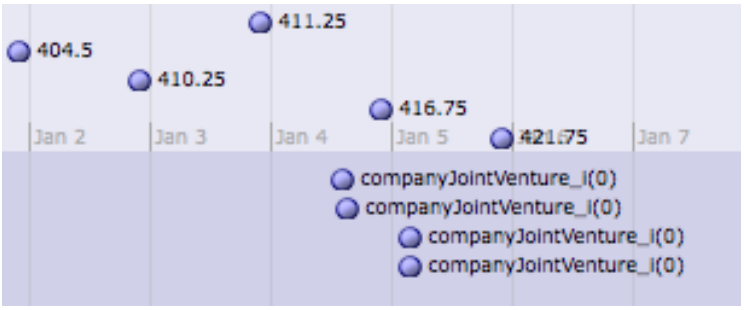

Fig. 4. Timeline with equity price and Joint Venture event

4.4 Experiment Limitations

There are several experiment limitations influencing the result of our tests. The most important limitation is the lack of data. Without news data, relations between objects cannot be evaluated. Also the quality of the annotations is a concern. By improving the event discovery process we are better off in understanding news information. Suppose that an event happens at the end of January and the repository doesn't contain news about that period; this could influence the price of equities even until February.

5 Conclusions and Future Work

This paper proposes a framework and a tool that detect dependency patterns between objects in economic news items. In a world in which news is retrieved in a split second, it is important to have a structured representation of news items so the interpretation can be done fast, either by machines or humans. We propose a timeline that makes it possible to sort data over time. Having data sorted over time, it is easy to analyze if the occurrence of an object is followed up by another object. For this purpose visualization can help in detecting dependency patterns. The tests we presented support this conclusion, by confirming our own visual observations with statistical tests.

We have to realize that experiment limitations influence the result of the tests. The lack of data, meaning not enough annotated news items with events, can lead to situations in which conclusions cannot be drawn. Another problem can be that news which is not about a specific company can still influence the equity price of that company.

As future work we would like to analyze object patterns in a larger context (than only by using Reuters news) to provide for a better understanding of the object relationships. For this we plan to use a multitude of news sources as well as different additional data types such as weather, natural disasters, macro-economic indicators, etc.

References

1. Apache Jena - A Free and Open Source Java Framework for Building Semantic Web and Linked Data Applications, http://jena.apache.org/
2. Distribution Tables. http://www.statsoft.com/textbook/sttable.html
3. FTSE 100/250. http://uk.finance.yahoo.com/
4. JQuery: a Javascript Quering Engine. http://www.jquery.com
5. Lucene Search Engine. http://jakarta.apache.org/lucene/docs/index.html
6. Prefuse Information Visualization Kit. http://prefuse.org/
7. Semlab. http://www.semlab.nl
8. Textmap.com. http://www.textmap.com
9. Timeline SIMILE Project. http://code.google.com/p/simile-widgets/
10. Bechhofer, S., van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D.L., Patel-Schneider, P.F., Stein, L.A.: OWL Web Ontology Language Reference. W3C Recommendation 10 February 2004 (2004)
11. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American 284(5), 34–43 (May 2001)
12. Blumberg, B., Cooper, D.R., Schindler, P.S.: Business Research Methods. McGraw-Hill (2005)
13. Chan, W.S.: Stock Price Reaction to News and No-news: Drift and Reversal After Headlines. Journal of Financial Economics 70(2), 223–260 (2003)
14. van Essen, M., Jongmans, M.: Thor: Creating a news visualization tool (2008)
15. Fitzpatrick, J.A., Reffell, J., Aydelott, M.: Breakingstory: Visualizing Change in Online News. In: Conference on Human Factors in Computing Systems 2003 (CHI 2003). pp. 900–901. CHI, ACM (2003)
16. Frasincar, F., Borsje, J., Hogenboom, F.: Personalizing News Services Using Semantic Web Technologies, chap. 13, pp. 261–289. E-Business Applications for Product Development and Competitive Growth: Emerging Technologies, IGI Global (2011)
17. Frasincar, F., Borsje, J., Levering, L.: A Semantic Web-Based Approach for Building Personalized News Services. International Journal of E-Business Research (IJEBR) 5(3), 35–53 (2009)
18. Geroimenko, V., Chen, C.: Visualizing the Semantic Web. Springer (2006)
19. Google Finance: http://finance.google.com
20. Klyne, G., Carroll, J.J.: Resource Description Framework (RDF): Concepts and Abstract Syntax. W3C Recommendation 10 February 2004 (2004)
21. Lloyd, L., Kechagias, D., Skiena, S.: Lydia: A System for Large-Scale News Analysis. In: 2th International Conference on String Processing and Information Retrieval (SPIRE 2005). Lecture Notes in Computer Science, vol. 3772, pp. 161–166. Springer (2005)
22. Pietriga, E.: IsaViz: a Visual Environment for Browsing and Authoring RDF Models. Eleventh International World Wide Web Conference (WWW 2002), Developer's day (2002)
23. Prud'hommeaux, E., Seaborne, A.: SPARQL Query Language for RDF. W3C Recommendation 15 January 2008 (2008)
24. Sintek, M.: OntoViz Tab: Visualizing Protégé Ontologies, http://protegewiki.stanford.edu/wiki/OntoViz