Doctoral symposium: Visualising complex event hierarchies using relevant domain ontologies

Riley Perry*
CSE, UNSW
rtpe380@cse.unsw.edu.au

Madhushi Bandara
CSE, UNSW
z5122242@cse.unsw.edu.au

Cat Kutay
FEIT, UTS
Cat.Kutay@uts.edu.au

Fethi Rabhi
CSE, UNSW
f.rabhi@unsw.edu.auu

ABSTRACT

With the growth of data available for analysis, people in many sectors are looking for tools to assist them in collating and visualising patterns in that data. We have developed an event based visualisation system which provides an interactive interface for experts to filter and analyse data. We show that by thinking in terms of events, event hierarchies, and domain ontologies, that we can provide unique results that display patterns within the data being investigated. The proposed system uses a combination of Complex Event Processing (CEP) concepts and domain knowledge via RDF based ontologies. In this case we combine an event model and domain model based on the Financial Industry Business Ontology (FIBO) and conduct experiments on financial data. Our experiments show that, by thinking in terms of event hierarchies, and pre-existing domain ontologies, that certain new relationships between events are more easily discovered.

CCS CONCEPTS

• **Human-centered computing** → *Visualization techniques*; *Visualization theory, concepts and paradigms*; *Empirical studies in visualization*;

KEYWORDS

Visualization

1 INTRODUCTION

With the growth of data available for analysis, people in many sectors are looking for tools to assist them in collating and visualising

---

*PhD Student

patterns in their material. We have developed an event based visualisation system which provides an interactive interface for experts to filter and analyse data. The system displays the occurrence of events over time on an annotated graph. Combinations of inbound events, and background knowledge are used to generate new events in real time. We believe that the framework and process presented is a unique method for building patterns in data.

The generated events are *complex events*. A data feed provides the initial events, which are called *simple events*. The process of creating new events, based on combining prior events, is Complex Event Processing (CEP). Using CEP hierarchies for Exploratory Data Analysis (EDA) [6] is first explored in [5]. EDA is a set of graphical methods and tools used to explore data.

Building on the ideas from [5] we add the concept of the use of a domain ontology, and domain knowledge. We show that this can assist users to tease out new concepts via the iterative visual development of the CEP hierarchies, and background knowledge from data that conforms to the ontology. These concepts, if applicable, are then made into event types themselves. The idea is to build a specific hierarchy around a particular need. The more atomic (single use) our event types get, the more visually we can see the logical hierarchy that leads to a particular event type being created. In the first section we explain the concept of complex events as used in present tools, then in the next section we introduce the new methods that relate these to ontologies. As a result this process of creating complex events we might think of novel, new event types by seeing patterns in the instances of new complex events appearing on a graph. To illustrate this process we provide the results from a running example of event processing with financial and news data, focused on stocks with a net profit of over that one billion USD.

The system architecture has been developed around the ontology in such a way that by adding another ontology the user can select complex events in the new domain. In this way the framework is generic and is not limited to financial analysis and could be tested in other domains such as network packet analysis, candidate data in the recruitment industry, etc.

2 COMPLEX EVENT PROCESSING

Visualising in terms of CEP event hierarchies can help finding unique patterns in data [5] and has been utilised in data anlysis tools previously. To generate complex events, we intially start with simple events from a data feed, which are *raw* events, a record of an occurrence in the real world. They are a record of some activity

within a system. CEP systems comprise of one or more *event sources* which provide the initial simple events. We will see we can create event hierarchies to model higher level concepts.

2.1 Aspects

According to Tukey [6], an event is defined by the following aspects: *Form*, the main attribute store for an event. *Significance*, the type of activity the event describes within its form. and *Relativity*, a series of events that caused the event.

When modelling an event in the visualisation the aim is to retain the ability to decompose back to the component events (stored in the relativity). Hence when large scale patterns are extracted the user can examine the attributes of the complex event.

2.2 Event Patterns

The purpose of defining *event patterns* is to use instances of them to find patterns in event data and generate new events based on these matches. Each new event that enters a system gets fed through a *match method* for each pattern instance. Depending on the outcome of the match method a new complex event may be generated.

2.3 Semantic Complex Event Processing

We are extending this concept of generating complex events to include more data to incorporate a wider range of sources and extract more features of events. This gives us a way of incorporating background knowledge into our queries, which often provides signicant information.

Adding rules and ontologies to CEP gives us Semantic Complex Event Processing (SCEP). This extends the logic of CEP event hierarchies to enable visualising SCEP based event hierarchies, which combine a broader range of data in the data analysis.

For example, a knowledge base query can filter out any simple events where the event does not relate to a car manufacturer with less that $10,000$ employees. In this example SCEP both affects an event's significance and form.

Event Stream [5]:

```
[(Name, BHP)(Price, 45)(Volume, 2000)(Time, 1)]
```

Semantic Knowledge Base (KB):

```
[(BMW, is_a, car manufacturer),
(car manufacturer, build, Cars),
(Cars, are built from, Metal),
(BMW, is a, Major_corporation),
(Major corporations, have, over 10,000 employees)]
```

In its simplest form we can add a knowledge base as a provider and then add a query (e.g. in SPARQL) in an event pattern match method which reference event attributes, ontology types, and domain data.

3 LINKING TO DOMAIN ONTOLOGIES AND FEEDS

To provide a generic visualisation tool, we provide a framework (shown in figure (1) that combines event hierarchies, a domain ontology, and background knowledge. *Ontologies* allow us to formally model, and reason about knowledge.
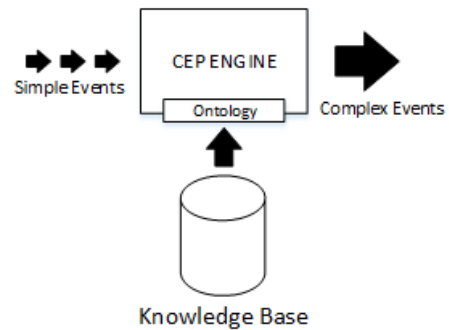

Figure 1: CEP + SCEP framework

The framework allows us to plug in ontologies and knowledge bases that conform to those ontologies. Queries to the knowledge base are done within the pattern's match method. The event instance(s), which are parameters to the match method, contain form attributes. The attributes and knowledge base details can be combined to generate new event instances of the pattern type.

There is research done in the finance domain such as [1] which propose ontology based approaches to correlate various activities such as correlation between news and financial instruments. The work in [2] represents the significance of ontology in linking different data sets and their causal relationships. The case study provided by [5] is an excellent example of the application of this research. In this new approach, we show that background knowledge that conforms to an ontology, and CEP, can be used for data exploration via a user interface. For this we need to find the appropriate ontology.

3.1 The Financial Industry Business Ontology (FIBO)

FIBO has a set of formal models that define unambiguous shared meaning for financial industry concepts born out of a requirement for common, shared meaning across data sources and message feeds in the financial services industry. By mapping any attributes of an event, or the event types themselves to FIBO entities we can utilise the domain knowledge to assist us with creation of our own event types and hierarchy. We use a cut down version of FIBO to retrieve data on companies.

3.2 Visualization of Events

We begin with a data feed. The data feed provides *Price Updates* and *News Updates* that are read in as simple events. When visualizing an event on a two-dimensional plane t (time) will usually be mapped to the x axis. There are two distinct classes of significance of simple event for visualisation purposes:

· *2D* - One of which will apply to both the x (time) axis and a single form attribute for the y axis. An example is a price update.
· *1D* - One of will apply to the x axis. An example is an individual news update.

The concept of a news story is an aggregate of news events that is based on an identifier form attribute (in our case this is a data attribute called a *PNAC*).

3.3   Level One (Simple) Events

Simple News Update Events: A simple *news update* event (NU) is an indivisible news related message. Each message consists of various codes including the *PNAC*. Simple news updates, when combined, form a partial or complete *News Story*. News stories, described next, are complex events.

Simple Price Update Events: A simple *Price Update* event (PU) contains the trading price, volume, and trading timestamp (or period) for a particular security.

3.4   Level Two Events

Events above level one are complex events. We describe here the type of complex events and how they are formed.

News Stories: A *News Story*, NS with news events $NU_1, NU_2, \ldots, NU_n$ can be represented by a simple tree diagram like this: NS sits in
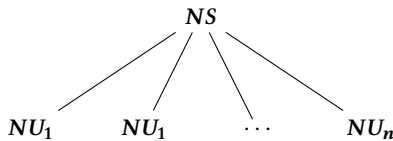
a two level *abstraction hierarchy*. $NU_1, NU_2, \ldots, NU_n$ all share the same *PNAC* but may be different occurrences of same news story. Attributes of NS are drawn or derived from this pool. CEP is a realtime technology used to make decisions quickly. A set of definitive Pattern rules for NS can be hard to determine as a story can span several days with many events, or just be comprised of an instantaneous single event.

Price Jump: is a *Price Jump* which contains positive price update events PU_1 and PU_2. An example of a price jump for the running
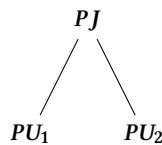
example is a price increase of over 2 points for a certain stock.

3.5   SCEP additions to patterns

SCEP give us two additional benefits when doing EDA. The first benefit is incorporating domain knowledge in pattern match methods. This can be useful when filtering out events that do not match certain criteria. Also domain knowledge can be used to populate attributes within the complex event being created. The second benefit of SCEP is the ability to see relationships between the data with can be used as a basis to create new event patterns.

The example in this paper uses a SCEP filter on the price updates which filters out all companies who have a net profit of less that one billion USD. The ontology is a simplified version of the FIBO ontology and the data comes from a custom data source which is based on a Thomson Reuters data feed.

4   VISUALISATION

We have developed and tested a prototype on financial data, looking at scenarios like price jumps and news stories. This allows us to link sentiment to price movements, as an example. The interface has allowed us to examine the patterns in the Complex Events, which have provided some initial results.

4.1   Improvement

The new user interface we have developed has 3 distinct sections. They are: An *Attribute Panel*, *Display Panel*, and a *Filter Panel*. The display panel is the main graph, where the root event sets up the x, y plane. Events are then stacked based on their Z-order (z). There is always a selected event, E_{sel}. The attributes in the attribute panel are those of the selected event. The filter panel allows the removal of events with attributes of certain values. Of particular interest is filtering out x, y ranges and also showing only events within the selected event's relativity.

When we want more information on the news story plus event we (for example) would simply select the individual event, then just the news story plus event and its constituent events would be shown. Attributes (in the attribute panel) that have been filled via the knowledge base are clickable and the ontology is shown graphically.
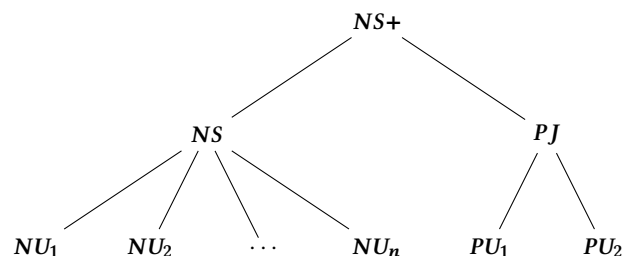
5   RESULTS

The process described has provided the underlying architecture for a visual interface that has enabled us to start extracting more complex event patterns. We provide here case studies of our work to illustrate how the pattern extraction works.

5.1   News Story Plus

We have higher level hierarchies of aggregation, when combined event stories with pricing, that can have attributes of their own. This allows us to build higher level concepts which can in turn be metrics for analysis and further aggregation

News Story Plus: A *News Story Plus* (NS_+) event is generated when there is a price jump within (and related to) the time period for a news story. Given a news story NS and a price jump PJ extra information can be gained from combining and doing calculations on the attributes of both. The concept of a news story plus (shown below) sits in three level CEP event abstraction hierarchy. Times-

tamps for $NU_1, NU_2, \ldots, NU_n$ always fall within the boundaries of PU_1 and PU_2, i.e.: $NU_1^{(t)} >= PU_1^{(t)}$ and $NU_n^{(t)} <= PU_2^{(t)}$. Preferably timestamps should be at regular intervals and should match exactly. i.e. $NU_1^{(t)} = PU_1^{(t)} \ldots NU_n^{(t)} = PU_2^{(t)}$.

**Figure 2: A higher level event that could be associated with negative sentiment**

It's useful at this point to introduce a variable, ϵ = event level-1, to represent the event level for display purposes. A news story plus for example has an ϵ of 2.

5.2 Easy Sentiment Analysis

Sentiment analysis aims to determine the attitude of a speaker or a writer with respect to some topic. For example positive text may contain words like "good wonderful, spectacular" and negative text might contain the words "bad, horrible, awful". We developed a hierarchy that utilised price jumps, and news stories. The experimental interface, detecting a negative sentiment and price fall, is shown in figure (2) just after a major price fall.

5.3 Story Slopes

Given a news story NS and a measure interval MI extra information can be gained from combining and doing calculations on the attributes of both. An example is a *News Story Slope* (NSS figure 3). The concept of a news story slope (shown below) sits in three level CEP event abstraction hierarchy.
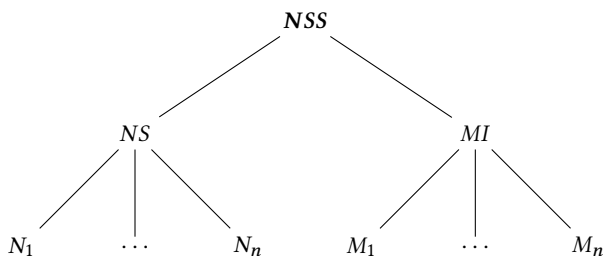
Timestamps for $N_1 \ldots N_n$, always fall within the boundaries of $M_1 \ldots M_n$, i.e.: $N_1^{(t)} >= M_1^{(t)}$ and $N_n^{(t)} <= M_n^{(t)}$. Preferably timestamps should be at regular intervals and should match exactly. i.e. $N_1^{(t)} = M_1^{(t)} \ldots N_n^{(t)} = M_n^{(t)}$.
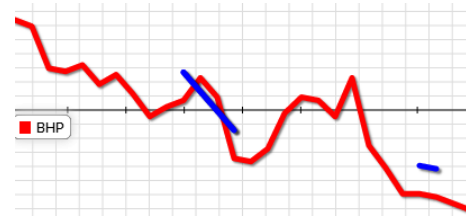

Figure 3: A News Story Slope - The blue lines are slope and magnitude

6 FUTURE DIRECTIONS

Any data that appears as a stream of events can be used by the framework presented in this paper. We are prospsing further tests in analysis of data the recruitment industry. Job applications can be seen as a series of simple events. Also Job rejections, interview outcomes, etc. could be seen as simple events.

The next stage is to run user studies to validate the framework's effectiveness in assisting users to extract patterns in the financial domain.

7 CONCLUSION

SCEP can be used for EDA in the financial domain to help produce new ideas for data analysis. By matching this to the FIBO we have developed a system where the data visualisation tools are generated from the domain ontology, we propose the system can be re-deployed in other domains.

More research should be done with other domains to verify the ability of the system to provide useful analysis with a variety of event attributes.

REFERENCES

[1] Kia Teymourian, Malte Rohde, and Adrian Paschke. 2012. Knowledge-based processing of complex stock market events. In Proceedings of the 15th International Conference on Extending Database Technology (EDBT '12), Elke Rundensteiner, Volker Markl, Ioana Manolescu, Sihem Amer-Yahia, Felix Naumann, and Ismail Ari (Eds.). ACM, New York, NY, USA, 594-597. DOI=http://dx.doi.org/10.1145/2247596.2247674

[2] Kia Teymourian. Malte Rohde, and Adrian Paschke. 2011. Processing of Complex Stock Market Events Using Background Knowledge. RuleML 2011 America Fort Lauderdale, Florida, USA.

[3] Sebastian Binnewies and Bela Stantic. 2012. OECEP: enriching complex event processing with domain knowledge from ontologies. In Proceedings of the Fifth Balkan Conference in Informatics (BCI '12). ACM, New York, NY, USA, 20-25. DOI=http://dx.doi.org/10.1145/2371316.2371322

[4] Opher Etzion and Jeffrey M. Adkins. 2013. Tutorial: why is event-driven thinking different from traditional thinking about computing?. In Proceedings of the 7th ACM international conference on Distributed event-based systems (DEBS '13). ACM, New York, NY, USA, 269-270. DOI=http://dx.doi.org/10.1145/2488222.2488271

[5] Perry R.T., Kutay C., Rabhi F. 2015. Using complex events to represent domain concepts in graphs. In: Kim K. (eds) Information Science and Applications. Lecture Notes in Electrical Engineering, vol 339. Springer, Berlin, Heidelberg

[6] Tukey, John (1977), Exploratory Data Analysis, Addison-Wesley.