

Frequent Pattern Trend Analysis in Social Networks

Puteri N. E. Nohuddin¹, Rob Christley², Frans Coenen³, Yogesh Patel^{1,3},
Christian Setzkorn², Shane Williams³

¹Department of Computer Science, University of Liverpool, UK,
puteri, frans@liverpool.ac.uk, ²School of Veterinary Science, University of
Liverpool and National Centre for Zoonosis Research, Leahurst, Neston, UK,
robc, c.setzkorn@liverpool.ac.uk ³Deeside Insurance Ltd., Deeside, UK,
yogesh, shane@deesideinsurance.co.uk

Abstract. This paper describes an approach to identifying and comparing *frequent pattern trends* in social networks. A frequent pattern trend is defined as a sequence of time-stamped occurrence (*support*) values for specific frequent patterns that exist in the data. The trends are generated according to *epochs*. Therefore, trend changes across a sequence epochs can be identified. In many cases, a great many trends are identified and difficult to interpret the result. With a combination of constraints, placed on the frequent patterns, and clustering and cluster analysis techniques, it is argued that analysis of the result is enhanced. Clustering technique uses a *Self Organising Map* approach to produce a sequence of *maps*, one per epoch. These maps can then be compared and the movement of trends identified. This *Frequent Pattern Trend Mining* framework has been evaluated using two non-standard types of social networks, the cattle movement network and the insurance quote network.

Keywords: Social Networks, Pattern Mining, Trend Mining, Trend Analysis.

1 Introduction

This paper introduces a social network Frequent Pattern Trend Mining (FPTM) framework. The framework includes a mechanism for identifying and extracting trends in social network data, grouping those trends and then identifying the “most interesting” trends. The input to the framework is a sequence of n time stamped social networks, partitioned into m sub-sequences, called *epochs*, of n/m time stamps each. A single trend in this context is defined in terms of a sequence of frequency count values associated with a particular pattern. An established frequent pattern mining algorithm, the Total-From-Partial (TFP) algorithm [13] [6] has been adopted to identify the frequent patterns, and modified so as to identify temporal sequences of frequent patterns (i.e. trends). In common with more general frequent pattern mining applications, a great many trends are typically identified. To limit the number of generated trends, constraints are applied to the nature of the frequent patterns. However, this only serves to

partially address the issue, in most case a significant number of trends remain. The remaining trends are thus clustered using a Self Organising Map (SOM) approach [1] so that similar trends are grouped. The application of the SOM technique results in a sequence of maps, one per epoch. These maps can then be compared; an “interesting” trend is defined as a trend whose location, with respect to the SOM nodes in which it is located, changes significantly from map to map.

The focus of the work is two specific social networks, the Deeside insurance quote network and the GB cattle movement network. These two networks are exemplars of two generic categories of social network, which the authors refer to as *star networks* and *complex star networks*. Further details of the nature and derivation of the networks are given in Section 3.

A formal definition of the trend mining problem, as envisioned in this paper, is presented in Section 4. A detailed review of the proposed Frequent Pattern Trend Mining (FPTM) framework is given in Section 5 together with a review of the components of the framework. A full evaluation of the framework is given in Section 6 and 7 using the cattle movement and Deeside networks respectively. The evaluation demonstrates that the framework can be used to identify interesting trends in social network data.

2 Previous Work

The proposed FPTM framework is founded on a number of technologies, namely frequent pattern mining and Self Organising Maps (SOMs). A brief review of both of these technologies is therefore provided in this section. The section is concluded with a review of some alternative approaches to trend mining related to the work described in this paper.

Frequent pattern mining is a well established technology that can be traced back to early work on Association Rule Mining (ARM) [4]. The original concept was to find all frequently occurring combinations of a global set attributes (the set of *frequent patterns*) that exist in a binary-valued data set and from these combinations extract relationships defined in terms of *association rules*. In the context of the work described in this paper, the authors are interested in the first part of this process, the identification of frequent patterns. Subsequent to the original ARM algorithm presented in [4] many alternative algorithms have been proposed. Examples include FP-growth [5] and Total-From-Partial [6] [7]. For the FPTM framework, the latter was adopted although similar fast frequent pattern mining algorithms could equally well have been adopted. To limit the number of frequent patterns discovered, a minimum support threshold is usually adopted. A pattern is only recorded as frequent if its frequency count is above this threshold. The lower the threshold, the greater the number of frequent patterns that are discovered.

The concept of Self Organising Maps (SOMs), a type of artificial neural network, was first introduced by Kohonen [2] [3]. A SOM is an effective visualization method to translate high dimensional data into a low dimension grid (map), with

$x \times y$ nodes. Initially, the SOM nodes are assigned with weight vectors which have the same dimensions as the input data. The nodes compete with each other to be stimulated to represent input data records according to a distance function¹ and a neighbourhood function². SOM have been used in many areas such as pattern recognition of optic nerve images [12], gene expression patterns [11] and manufacturing processing control [14].

There has been some work on social networks trend analysis. Gloor *et.al.* introduced a novel trend analysis algorithm to generate trends from Web resources [10]. The algorithm calculates the values of temporal *betweenness* of online social network node and link structures to observe and predict trends on the popularity of concepts and topics such as brands, movies and politicians. Research in social networks trend mining has provided advantages for online viral marketing [8], stock market activities [9] and many more. There has been some work on the identification of trends in time stamped sequences of binary valued (ARM) data sets. Early examples can be found in work on Jumping and Emerging Patterns. Emerging Patterns describe patterns with support counts change between time stamps [16]. Jumping Patterns describe patterns whose support counts change drastically (jump) from one time stamp to another. In medical research, jumping and emerging pattern trends have been used to monitor the progress of cancer cells [17]. The concept of frequent pattern trends defined in terms of sequence of frequency counts has also been adopted in [18] in the context of longitudinal patient datasets. In [18], trends are categorised according to pre-defined prototypes, in the authors' work, they are categorised using SOM technology.

3 Target Social Networks

The work described in this paper is directed at two specific social networks. The first is a customer network extracted from an insurance company's database referred to as the Deeside insurance quote network. The second was extracted from the Cattle Tracing System (CTS) database in operation in Great Britain (GB) and is referred to as the GB Cattle Movement Network. The generic nature of these networks is presented (in a stylised form) by the two "network snapshots" given in Figures 1 and 2. With reference to Figure 1, the network is characterised by a single "star shape" with all nodes communicating with one *super-node*, the authors refer to this type of network as a *star network*. Note that, as shown in the figure, not all network nodes will be necessarily communicating (linking) with the super-node at any given time stamp. The generic network snapshot given in Figure 2 is a more complex version of that given in Figure 1, and the authors refer to this as a *complex star network*. The network is characterised by a number of disconnected "star" sub-networks of varying size. Again, not all network nodes (with respect to the snap-shot time stamp) are necessarily communicating (linking) with any of the other nodes. Note also that, some of the "star" sub-networks comprise only two nodes. The derivation of both networks

¹ Euclidean function is the distance function used to calculate shortest distance.

² Gaussian function is used to determine the neighbourhood size on the map.

is briefly described in the rest of this section. In both cases, the authors have applied the FPTM framework to identify the interesting trends that feature in a sequence of time stamped snapshots of these networks.

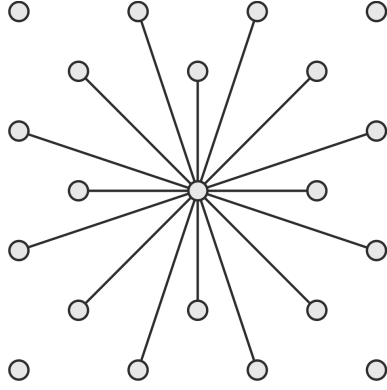


Fig. 1: (Stylised) Star Network

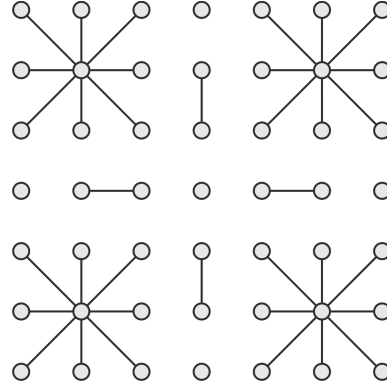


Fig. 2: (Stylised) Complex Star Network

The Deeside insurance quote network was extracted from a sample of records taken from the customer database operated by Deeside Insurance Ltd. (collaborators on the work described in this paper). Twenty-four months of data were obtained comprising, on average, 400 records per month. In total, the data set comprised 250 records with (after discretisation and normalisation) 314 attributes. The data was processed to produce a sequence of 24 networks, one per month; divided into two epochs comprising 12 months each, 2008 and 2009. In the resulting networks, the nodes comprised postal areas (characterised by the first few digits of UK post/zip codes), and the links are the number of requests for specific types of insurance quotes received for the given time stamp. Thus, although not shown in Figure 1, there can be many links between the super node and the outlying nodes. An example of a type of insurance quote, attached to a link, might be:

$$\{NumberOfQuotes = \{10 : 20\}, CarType = Vauxhall, EngineSize = \{1500 : 1999\}, OffenceCode = SP, Fine = \{200 : 300\} \text{ and } Gender = Male\}$$

where: the value $\{10:20\}$ indicates the *NumberOfQuotes* is in a range between 10 and 20; the value $\{1500:1999\}$ states the *EngineSize* is within the range 1500 and 1999, the value SP is an *OffenceCode* indicating exceeding the speed limit, and the value $\{200:300\}$ indicates a *Fine* of between 200 and 300. The number of quotes attached to a specific link ranged between 10 and 40. The average number of nodes in a single (one month) network was 120, and the average number of links 200.

The cattle movement network was extracted from the Cattle Tracing System (CTS) database in operation in Great Britain (GB). The CTS database

was introduced in September 1998 and updated in 2001 as a result of a number of outbreaks of bovine diseases. The database is maintained by DEFRA, the Department for Environment, Food and Rural Affairs, a UK government department. The database records all cattle movements in GB, each record describes the movement of a single “beast”, identified by a unique ID number, between two *holding locations* (farms, markets, slaughter houses, etc.). However, the database can be interpreted as a social network, where each node represents a geographical location and the links the number of beasts moved between locations. In a similar manner to the Deeside insurance quote network, the links describe specific types of cattle movement, thus there could be more than one link between pairs of nodes. An example of a type of cattle movement that might be attached to a link is:

$$\{NumberOfBeasts = \{50\}, BeastType = Limousin\ Cross, AnimalAge = \{1 : 6\}, PTI = 4\ and\ Gender = Female\}$$

where the attribute label *PTI* is the Parish Testing Interval which describes the frequency of disease detection testing for each node; the value is between 1 and 4 years. The number of beasts attached to the link is 50. Each node describes a location defined in terms of 100km grid squares. Four years worth of data, from 2003 to 2006, were used to generate the networks giving a sequence of 48 networks, one per month, divided into four epochs of 12 months each. After discretisation and normalisation, the average number of nodes within a single network was 150,000 and the average number of links was 300,000 with 445 attributes. It should be noted that some nodes in the network, represent zoos or “hobby farms”, which seldomly move beasts to other locations.

4 Problem Definition

The FPTM framework is directed at finding interesting trends in social network data. A trend is deemed to be interesting if its “shape” changes significantly between epochs. An epoch is defined in terms of a start and an end time stamp. The input to the FPTM framework is a sequence of n time stamped datasets $D = \{d_1, d_2, \dots, d_n\}$ partitioned into m epochs (note that n should typically be some multiple of m). Each data set d_i comprises a set of records such that each record describes a social network node pairing, the description consists of some subset of a global set of attributes A that describes the network (some example records were given in Section 3). There are $2^{|A|} - 1$ patterns that may exist in any given dataset. The *support* (s) for a pattern I in a dataset d_i is the number of occurrences of the pattern in d_i expressed as a percentage of the number of records in d_i . A trend (t) with respect to a pattern I is then a sequence of support values associated with the data sets represented in an epoch, $t = \{s_1, s_2, \dots, s_{n/m}\}$. Each time stamp (i) therefore has a set of trends (T_i) associated with it. The complete set of trends is then given by $T = \{T_1, T_2, \dots, T_n\}$. The set of trends associated with an epoch E_j (where $0 < j \leq m$) is then defined as $E_j = \{T_{1+k}, T_{1+(k+1)}, \dots, T_{1+(k+(n/m)-1)}\}$ (where

$k = (j - 1)(n/m)$. The complete set of trends, in a sequence of epochs E , is then given by $E = \{E_1, E_2, \dots, E_m\}$. The objective is then to identify interesting individual trends lines that exist across the set E . We have already indicated that “interestingness” is defined according to the distance a trend line moves between SOM maps. A more precise definition of interestingness will be given in the following section, Section 5.

5 The Frequent Pattern Trend Mining (FPTM) Framework

Figure 3 gives a schematic of the FPTM framework. The process commences in the top left of the figure with the collection of datasets $D = \{d_1, d_2, \dots, d_n\}$ describing a sequence of n social networks, each associated with a discrete time stamp of between 1 and n . During preprocessing, constraints may be applied to the nature of the frequent item sets that the authors may be interested in. The constraints take the form of feature-value pairs (attributes) that must be included in a frequent pattern for it to be considered relevant.

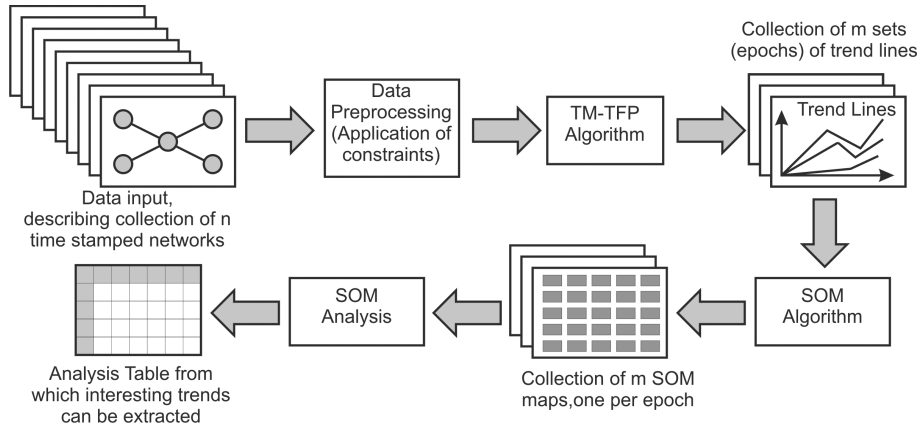


Fig. 3: Schematic illustrating the operation of the FPTM framework

The next stage is to obtain the set of trends T . This is achieved, as noted above, using an extension of the TFP algorithm called the the TM-TFP (Trend Mining TFP) developed by the authors. The advantage of TFP is that it is very efficient, this is facilitated by the use of two “set enumeration tree” style storage structures, namely the P-tree and the T-tree. The P-tree is used to summarise the input data and, as a result of its construction process, stores partial support counts. The T-tree then stores the frequent item sets contained in the input dataset and is generated from the P-tree. The T-tree can more accurately be described as a *trie*, in that levels in the T-tree branches are actually vectors (arrays). This then facilitates fast “look-up”. TM-TFP was built on-top of TFP and

includes a TM-tree to store *trend lines* (sequences of support values associated with individual patterns with respect to specific time stamps and consequently specific epochs). The output from TM-TFP is a set of trend lines associated with the given set of epochs $E = \{E_1, E_2, \dots, E_m\}$. In situations where a pattern is sometimes frequent (above the support threshold) and sometimes infrequent, a value of 0 is recorded where the support count falls below the threshold. The authors did experiment with the use of negative borders, as advocated in [15] (amongst others), however it was quickly realised that it was easier to simply use a relatively low support threshold (thresholds of 1, 2, 3, 4 and 5 were used for the experiments reported in this paper).

The next stage is to categorise the trends using a Self Organising Map (SOM) algorithm, one map was created per epoch. The process commenced with the generation (training) of a prototype $x \times y$ node map such that each node would represent a category of trend. The authors experimented with different mechanisms for training the SOM, including: (i) devising specific trends to represent individual nodes; (ii) generating a collection of all trends that are arithmetically possible and training the SOM using this set; and (iii) using some, or all, of the trends in the first epoch to be considered (E_1). The first required prior knowledge of the trend configurations in which the authors might be interested, as the objective was to discover these trends it seemed counter intuitive to first try and guess what they might be. With respect to the second option, it was discovered that this resulted in a map where the majority of nodes, once the maps had been populated, were empty. The third option was therefore adopted, the SOM was trained using the trend lines associated with the first epoch. The resulting *prototype map* was then populated with data for the remaining E_2 to E_m epochs to produce a sequence of m maps.

Temporal trend analysis was then applied to the sequence of m maps. The aim was to compare individual trends so as to discover significant temporal trend changes. Temporal trend changes are defined in terms of the “distance” a trend moves between two subsequent SOM maps. This is encoded in a matrix structure that holds “distances moved” for all trends and all sequences of pairs of SOM maps. Note that, some trends may remain in the same node for the entire sequence of m maps. Some other trends may fluctuate between nodes. While other trends may slowly move across the m maps. The greater the distance moved, the more interesting the trend is deemed to be.

6 Analysis Using The Deeside Insurance Quote Social Network

In this, and the following section, an analysis of the proposed FPTM framework is presented in the context of the Deeside insurance quote network (this section), and the GB cattle movement network (following section). In both cases, the analysis is conducted by considering the output from each stage and the nature of the discovered interesting trends.

For the analysis using the Deeside network, the authors applied a number of pattern constraints so as to reduce the number of patterns to be considered. The experiments used 3 pattern constraints with the selected attribute-value pairs:

$$\begin{aligned} \text{Constraint 1: } & \{DriveAge = \{24 : 40\}\} \\ \text{Constraint 2: } & \{Gender = Male\} \\ \text{Constraint 3: } & \{PostcodeArea = CH\} \end{aligned}$$

Constraint 1 has the effect of insisting that frequent patterns include the attribute $DriveAge = \{24 : 40\}$, while Constraint 2 has the effect of limiting the set of frequent patterns to those where $Gender$ has the value $Male$. Constraint 3 has the effect of restricting patterns to those that include the $PostcodeArea$ “CH” (Chester).

Table 1 gives the number of trends discovered using all the data; and with the application of Constraint 1, Constraint 2 and Constraint 3. With low support thresholds, a large number of trends are generated in each case. When a constraint is imposed, the number of records to be considered is naturally reduced therefore fewer trends are discovered. Also as the support threshold increases, the number of trends decreases. Figure 4 presents the prototype SOM for trends generated using a support threshold 1% with Constraint 1 ($\{DriveAge, \{24 : 40\}\}$). The prototype map displays the characteristics of clusters of trend lines. With reference to the figure, node 1 (top-left) represents trends with high support from January to March, while node 18 (center) portrays trends with fluctuating support values in April, June and August. Note that, the distance between nodes indicates the dissimilarity between nodes; the greatest dissimilarity is thus between nodes at opposite ends of the diagonals. Based on this prototype map, a sequence of m maps, in which all trends for E_1 to E_m are fitted into, are generated.

In the next stage of the FPTM process, the trend and node details from the m maps are transferred into matrix for trend analysis. Table 2 shows examples of trends (representing frequent patterns), with Constraint 1, that migrated from SOM m_1 to m_2 . For example, the trend line representing the pattern: $\{EngineSize = \{\leq 1000\}, CarType = Nissan, DriverAge = \{26 : 30\}\}$ was in cluster node 5 (middle top in Figure 4) in 2008, but moved diagonally to node 49 in 2009. The trend shape has changed significantly so it is marked as an interesting trend.

Table 1: Number of identified trends using Deeside insurance quote network

Support Threshold (%)	No Constraint	Constraint 1	Constraint 2	Constraint 3
1	830306	8239	5621	3965
2	206219	2163	1431	1595
3	94369	1038	677	863
4	55445	669	401	563
5	37239	469	283	427

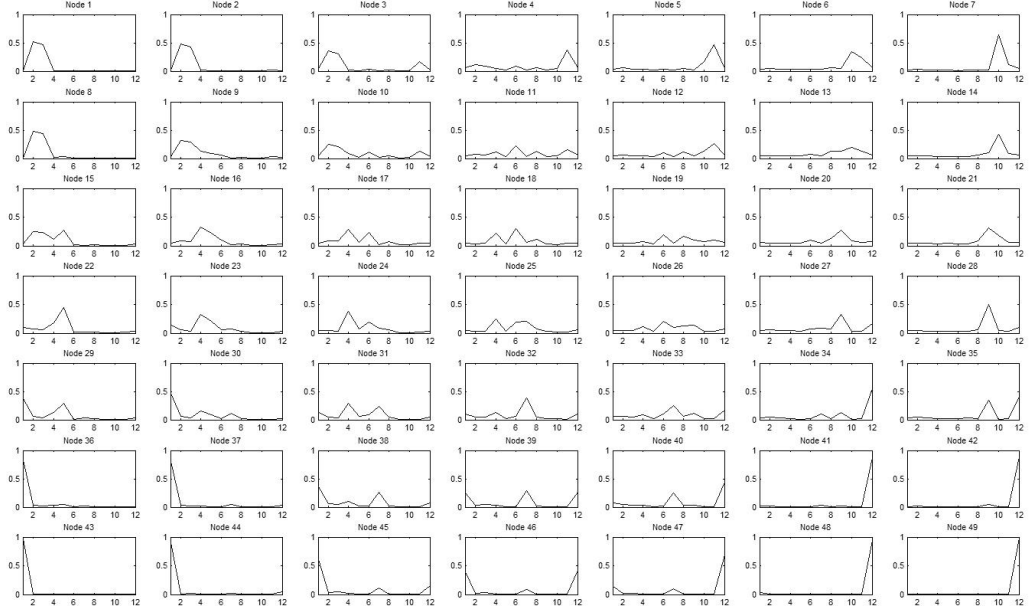


Fig. 4: Deside Prototype Map for trends with Constraint1

7 Analysis Using GB Cattle Movement Social Network

For the analysis using the GB cattle movement network, the authors also applied several pattern constraints. In the reported experiments, 3 pattern constraints were applied:

$$\begin{aligned}
 \text{Constraint 1: } & \{BreedType = Beef\} \\
 \text{Constraint 2: } & \{BreedType = Dairy\} \\
 \text{Constraint 3: } & \{SenderLocationType = \\
 & \text{Agricultural holdings, ReceiverLocationType} = \text{Market}\}
 \end{aligned}$$

Again, Constraint 1 and Constraint 2 filter records with beasts used for *Beef* or *Dairy*. Whereas, Constraint 3 selects movement records from *Agricultural holdings* to the *Markets*.

Table 3 presents the number of trends discovered using all the available data and with 1, 2 and 3 applied respectively constraints. As in the case of the Deeside data, as the support threshold increases, the number of trends decreases. Figure 5 illustrates the prototype SOM for trends generated with support threshold 1% with Constraint 1, ($\{BeastType = Beef\}$). The prototype map displays node clusters of the discovered CTS trends. For example, node 1 (top-left) represents trends that have high support in early summer (May), while node 43 (bottom-left) indicates trend lines with high support in autumn only (October). Again,

Table 2: Examples of Deeside trends migrating from one SOM node to another (*Dist* = distance value)

Example Trends	Node 2008	<i>Dist</i>	Node 2009
$\{EngineSize = \{\leq 1000\}, CarType = Toyota\}$	49	2.0	35
$\{EngineSize = \{\leq 1000\}, CarType = Toyota, DriverAge = \{\geq 51\}\}$	49	3.0	28
$\{EngineSize = \{\leq 1000\}, CarType = Nissan\}$	26	1.41	34
$\{EngineSize = \{\leq 1000\}, CarType = Nissan, DriverAge = \{26 : 30\}\}$	5	6.32	49

based on this prototype map, a sequence of m maps, in which all trends from E_1 to E_m are fitted into, was generated.

Table 4 presents some examples of trends, with Constraint 1, that migrated from m_1 to m_4 . For example, the trend line representing the frequent pattern: $\{ReceiverArea = 24, SenderLocType = Algricultural\ holdings, BeastType = Chianina\}$ was in cluster node 47 in 2003 and moved to node 15 in 2004, but then migrated to node 8 in 2005 and node 48 in 2006. The greater distance values in m_3 (2005) and m_4 (2006) meant that this trend was deemed to be interesting.

Table 3: Number of identified trends using GB cattle movement network

Support Threshold (%)	No Constraint	Constraint 1	Constraint 2	Constraint 3
1	25736	2333	2583	1195
2	8945	1019	1181	535
3	4393	541	715	383
4	2739	349	483	311
5	1928	249	339	267

Table 4: Examples of CTS trends migrating from one SOM node to another (*Dist* = distance value)

Example Trends	Node 2003	<i>Dist</i>	Node 2004	<i>Dist</i>	Node 2005	<i>Dist</i>	Node 2006
$\{ReceiverArea = 24, SenderLocType = Algricultural\ holdings, BeastType = Chianina\}$	47	5.66	15	1.0	8	7.07	48
$\{ReceiverArea = 24, SenderLocType = Algricultural\ holdings, BeastType = Lincon\ Red\}$	26	0.0	26	4.24	2	2.82	18
$\{ReceiverArea = 24, SenderLocType = Algricultural\ holdings, BreedType = Beef\}$	26	0.0	26	3.60	9	3.60	26
$\{ReceiverArea = 24, SenderLocType = Algricultural\ holdings, BreedType = Beef, BeastType = Chianina\}$	47	5.66	15	1.0	8	7.07	48

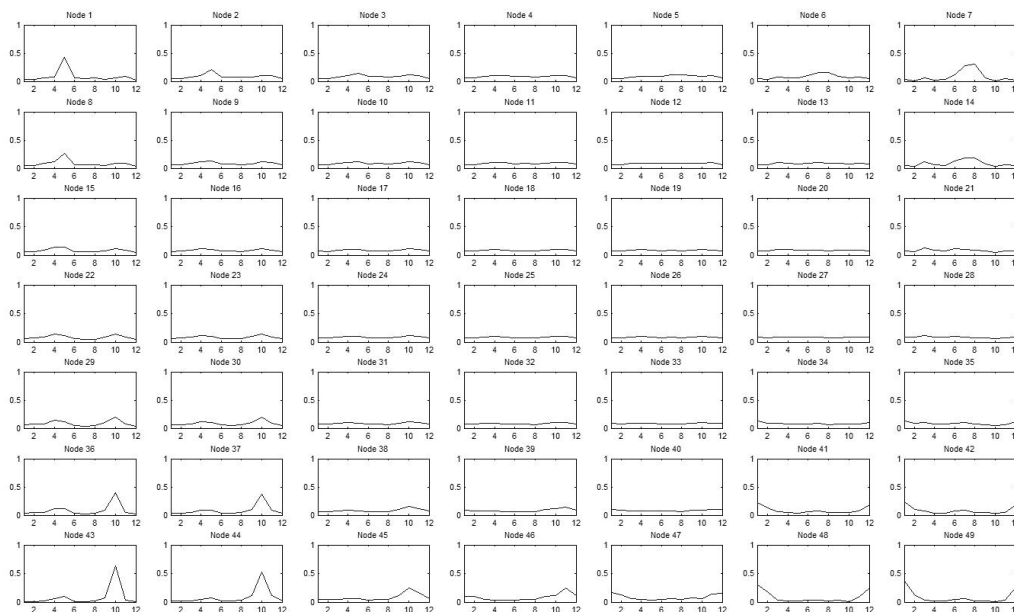


Fig. 5: CTS Prototype maps with Constraint 1

8 Conclusions

In this paper, the authors have described the FPTM framework for identifying interesting trends in social network data. The trends are defined in terms of sequences of support counts associated with individual patterns across a sequence of time stamps associated with an *epoch*. Interesting trends are defined as those whose “shape” changes across two or more epochs. Shape changes are captured by plotting identified trends on a sequence of SOM maps, one per epoch, and determining the distance individual trends move between subsequent maps. The greater the distance the more interesting the trend is deemed to be. The operation of the FPTM framework was illustrated using a “star” social network and a “complex star” social network, more specifically the Deeside insurance quote network and the GB cattle movement network which have formed the focus of this work. Future work that the authors intend to undertake is on how to apply the technique to other forms of data than social network data. In particular, the authors are looking to apply the technique to longitudinal patient data.

References

1. Cottrell, M., Rousset, P.: A powerful Tool for Analyzing and Representing Multidimensional Quantitative and Qualitative Data. In Proceedings of IWANN 97. LNCS, vol. 1240, pp. 861-871. Springer Berlin, Heidelberg (1997)
2. Kohonen, T.: The Self Organizing Maps. Neurocomputing 21 (1-3), 1-6 (1998)

3. Kohonen, T.: The Self Organizing Maps. Volume 30 of Springer Series in Information Sciences. Springer, Berlin, Heidelberg (2001) (Third Edition 2001)
4. Agrawal, R., Imielinski, T., and Swami, A.: Mining Association Rules between Sets of Items in Large Databases. In Proceedings of ACM SIGMOD Conference, pp 207-216 (1993)
5. Han, J., Pei, J. and Yiwen, Y.: Mining Frequent Patterns Without Candidate Generation. Proceedings ACM-SIGMOD International Conference on Management of Data, ACM Press, pp 1-12 (2000)
6. Coenen, F.: The LUCS-KDD TFP Association Rule Mining Algorithm, Department of Computer Science, The University of Liverpool, UK (2004)
7. Coenen, F., Leng, P. and Ahmed, S. : Data Structures for association Rule Mining: T-trees and P-trees. IEEE Transactions on Data and Knowledge Engineering, vol 16, no 6, pp 774-778 (2004a)
8. Richardson, M. and Domingos, P.: Mining Knowledge Sharing Sites for Viral Marketing, Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pp 61 - 70 (2002)
9. Choudhury, M.D., Sundaram, H., John, A. and Seligmann D.D.: Can blog communication dynamics be correlated with stock market activity? Proceedings of the nineteenth ACM conference on Hypertext and hypermedia, pp 55-60 (2008)
10. Gloor, Peter A, Krauss, Jonas S., Nann, Stefan, Fischbach, Kai and Schoder, Detlef, Web Science 2.0: Identifying Trends Through Semantic Social Network Analysis. Social Science Research Network (2008)
11. Wang, J., Delabie, J., Aasheim, H.C., Smel, E., Myklebost, O.: Clustering of the SOM easily reveals distinct gene expression patterns: results of a reanalysis of lymphoma study. BMC Bioinformatics, vol 3(36) (2002)
12. Yan, S., Abidi, S.R., and Artes, P.H.: Analyzing Sub-Classifications of Glaucoma via SOM Based Clustering of Optic Nerve Images, Proceedings of MIE2005 - The XIXth International Congress of the European Federation for Medical Informatics, pp 483-488 (2005)
13. Coenen, F.P., Goulbourne, G., Leng, P.: Computing Association Rules Using Partial Totals. Principles of Data Mining and Knowledge Discovery. LNCS, vol. 2168, pp. 54-66. Springer Berlin, Heidelberg (2001)
14. Kohonen, T., Oja, E., Simula, O., Visa, A., Kangas, J.: Engineering applications of the Self-Organizing Map. Proceedings of the IEEE, vol. 84(10), pp. 1358-1384 (1996)
15. Toivonen, H.: Sampling Large Databases for Association Rules, Proceedings of the 22th International Conference on Very Large Data Bases, pp 134 - 145 (1996)
16. Dong, G., and Li, J.: Efficient Mining of Emerging Patterns: Discovering Trends and Differences. In Proceeding of fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 43-52 (1999)
17. Yu, L., Chung, F., Chan, S., and Yuen, S.: Using Emerging Pattern Based Projected Clustering and Gene Expression Data for Cancer Detection. 2nd Asia-Pacific Bioinformatics Conference, New Zealand (2004)
18. Somaraki, V., Broadbent, D., Coenen, F. and Harding, S.: Finding Temporal Patterns in Noisy Longitudinal Data: A Study in Diabetic Retinopathy. To appear, Proc. ibia Industrial Conf. on Data Mining, Springer LNAI. (2010)