A Recommender System Architecture for Predictive Telecom Network Management

Faisal Zaman, Gabriel Hogan, Sven van der Meer, *Member, IEEE,*
John Keeney, Sebastian Robitzsch, and Gabriel–Miro Muntean, *Member, IEEE*

Abstract—Current telecom networks generate massive amounts of monitoring data consisting of observations on network faults, configuration, accounting, performance and security. Due to the ever increasing degree of complexity of networks, coupled with specific constraints (legal, regulatory, increasing scale of management in heterogeneous networks), the traditional reactive management approaches are increasingly stretched beyond their capabilities. A new network management paradigm is required that takes a pre-emptive rather than a reactive approach to network management.

This work presents the design and specification of E-Stream, a predictive recommendation based solution to automated network management. The architecture of E-Stream illustrates the challenges of leveraging vast volumes of management data to identify pre-emptive corrective actions. Such design challenges are mitigated by the components of E-Stream, which together form a single functional system. The E-Stream approach starts by abstracting trace information to extract sequences of events relevant to interesting incidents in the network. After observing event sequences in incoming event streams, specific appropriate actions are selected, ranked and recommended to pre-empt the predicted incidents.

Index Terms—Telecommunications, Network, Management, Predictive, Recommender, Architecture.

1 Introduction

The combined effect of the increase in users and communicating devices, demand for service quality and diversity, support for mobility, and desire for social connectedness and communication has driven unprecedented and exponential growth in telecoms network management data. Following this growth of users and devices, by 2020 the total number of connected devices will reach up to 50 Billion [1]. Consequently, the number of Network Elements (NEs) to manage will increase significantly. The amount of network management data transmitted from these NEs is expected to be at Exabyte levels. Also as heterogeneous networks are becoming a reality with the deployment of micro-, femto- and pico-cells [1] the complexity of the O&M tasks [2] scales-up accordingly. To maintain such complex networks current O&M approaches need to be extended in order to provide efficient and high quality communication services to end users. This impacts mostly on operating costs for operators as today's approaches for monitoring rapidly expanding user and device volumes will require a significant increase in management personnel, which based on current approaches is economically unsustainable. Many of the tasks required of human network managers are repetitive and involve wading through huge amounts of monitoring data. A new network management paradigm is required that is capable of automating the monitoring and repetitive tasks, and most importantly leverage massive volumes of network trace information to deploy a pre-emptive rather than reactive approach to predict issues and suggest timely appropriate remedial or preventative actions for network management.

More automated approaches are required to assist Network Operations Centre (NOC) op-

- *F. Zaman, S. Robitzsch & G.-M. Muntean are with the Network Innovation Centre, RINCE, Dublin City University, Ireland. E-mail: zaman.faisal@ieee.org@dcu.ie*
- *G. Hogan is with Centre for Global Intelligent Content (CNGL)*
- *S. van der Meer & J. Keeney are with the Network Management Lab at LM Ericsson Ireland.*

Manuscript received June 15, 2013

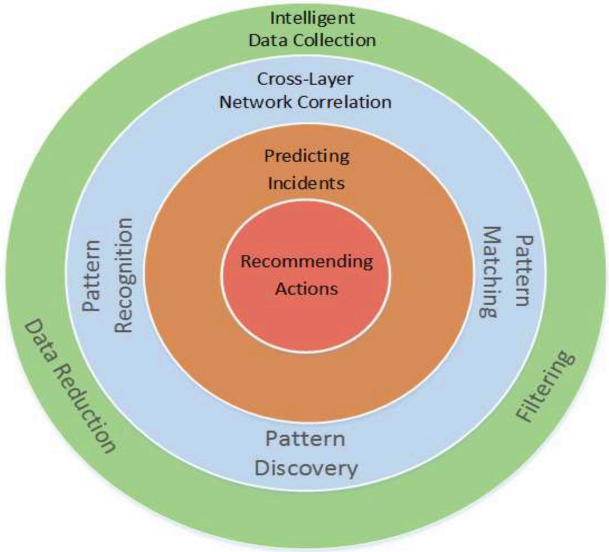


Fig. 1. E-Stream components in different information processing layers.

erators to manage complex network operation scenarios. Intelligent techniques with the capability to decipher the *recurrent* nature of predictable network incidents can unravel the link between predictive symptoms and the occurrence of a particular network scenario. Moreover, this insight into patterns of symptomatic events can be leveraged to prescribe potential solution(s) for particular scenarios. Integrating these two functionalities into a single information processing system is proposed in E-Stream. E-Stream analyses the likelihood of occurrence of potential network incidents and recommends the most appropriate solution for the incident. In addition to this, E-Stream gives the human operator control to adopt decisions while allowing the system to learn the decision recommendations over time and adapt and evolve by assimilating response know-how from the human expert.

From an operational point of view, E-Stream takes the network traces as inputs and transforms the massive volume of information into simple prescriptive actions as the outputs. First, E-Stream discards unnecessary, redundant and noisy information in order to observe patterns in the occurrence of the network incidents. Patterns are then screened, indexed and associated with the relevant network solutions. Finally, the "pattern-solution" templates of similar incidents are utilised to suggest proper corrective actions for network scenarios similar to those in which similar patterns were previously observed. The performance of the components at each phase is scaled by ingesting the incoming traces in an adaptive way, distributing the processing resources, and parallelizing the computational tasks.

Autonomic Network Management (ANM) [3] emerged as an approach to overcome the ever increasing complexity of network management within the Fault, Configuration, Accounting, Performance, Security (FCAPS) framework. Efficient Fault Management (FM) techniques proposed as part of an Autonomic Network Management System (ANMS) should be able to progressively learn and identify network faults but do not present solutions [4]. A similar approach [5] for pre-emptive detection of critical events in the area of service management proposed a process to discover potential predictive patterns in the log files to detect the occurrence of upcoming faults. Network management based on preventative maintenance and statistical process control (widely used in manufacturing industries) was proposed in [6]. However, the E-Stream approach for predictions and recommendations is innovative in two main aspects. First, it supports processing events from heterogeneous sources, and in this way overcomes classic problems of FCAPS silos. Second, E-Stream addresses the requirements of the Telecommunication Management Network (TMN) in terms of scale and performance. E-Stream combines both aspects into a holistic system design and implementation which addresses some of the challenges the telecommunication industry is facing today. Additionally the process of providing corrective actions based on the predicted network incidents, and with minimal human supervision, is novel and very different than the approach of gradually learning about the network faults in ANMS.

This work is the result of a collaborative Dublin City University-LM Ericsson Ireland project which integrates results of data mining, predictive analytics and recommender systems research.

2 E-STREAM CHALLENGES

The major challenge in designing E-Stream is the complex granular structure inherent in network data; such complex structures in the data complicate efforts to automatically find meaningful information from the data. The task is to transform the incoming information through different processing layers and finally deliver the recommendations. It requires expert knowledge on the deployment and management of the resources in each layer. To do so E-Stream is composed of several components and each component addresses and mitigates the challenges in processing the information and transferring it to the next layer. In Figure 1 the components of the E-stream architecture are placed in appropriate processing layers. The challenges addressed by each component in each layer are also indicated.

2.1 Intelligent Data Collection

Stream of network traces (denoted as "e-streams") originate from the telecom networks often arrive at extremely high rates straining the I/O and computational ability of the system. The traces consist of information patterns that can be correlated to the underlying network behaviour. E-streams are generated from various sources, and report on a huge variety of parameters, operations, warnings, and faults, only some of which is relevant for any particular management use-case. Combined with a potential for event storms, this in turn makes it very difficult to extract meaningful knowledge from the data within a limited time duration.

Data Dimension Reduction processes aim to reduce the volume of data being ingested by identifying and removing noise events, low importance events, and periodic or repetitive events. Dimension reduction processes therefore aim to identify the most important events, and data within those events. Simple data reduction techniques can serve the purpose of controlling the data ingestion, but it should be capable of handling sizeable volume of data. Parallelising the reduction process can address this problem, but raises the possibility more approximation errors. Event-based Stream Processing (ESP) techniques holds most promise for parallelised event reduction processes with low latency, but care must be taken to maintain scalability [7].

2.2 Cross-Layer Network Correlation

Network traces (e-streams) are not static and demonstrate burstiness, jitter, delay (out of order arrival) and data loss. In order to explore e-streams, the underlying individual sources need to be analysed to discover, recognise and match patterns across all the sources. These patterns can be indicative of correlative scenarios that are difficult to decipher from individual sources. Pattern discovery is required to detect the existence of patterns which have not been previously observed. Pattern matching in the event stream is required to indicate the occurrence of previously observed patterns. Pattern recognition is required in order to determine the likelihood of a candidate pattern becoming an exact match and to allow the prediction of and therefore the prevention of an incident occurrence.

For event-based pattern matching, Complex Event Processing (CEP) is a recognized technique. Topologically-aware reasoning (TAR) addressed the problem of discovering and matching patterns to identify the network faults based on spatio-temporal patterns [8]. Automated profiling of network events by modelling the event sequences is beneficial for pattern matching [9].

2.3 Predicting Incidents

Predicting incidents is dependent on pattern recognition accuracy, i.e. the probability of correctly identifying a complete pattern from first symptom to incidence occurrence. Incidents can be categorised at a high level with the following characteristics: incidents occur frequently or rarely in time; incidents are simple or complex; incidents have a simple or complex resolution. System behaviour (event-patterns) characterising an incident can present as a small number of symptoms in a single data set or as a very large number of symptoms across many data sets. Predicting incidents therefore ranges from those which occur very often and have very simple analysis to those which occur rarely and involve very complex analysis. Temporal analysis is constrained by the *amount* of observation that can be supported in any given time period, i.e. the available time for observation is inversely proportional to the volume of data being observed. The recognition accuracy is constrained by the number of candidate (probable) patterns being observed.

2.4 Recommending Actions

Action recommendation is dependent on a number of factors including context, audience, existing action/responses, and validation.

Context: In a typical telecommunication system when an incident occurs, one of a number of formal and prescribed responses to the incident is normally followed. This typically has instructional and a procedural aspects, i.e. the specific tasks to be carried out, the order or priority of the action/response and the reporting and tracking of the incident through for instance trouble ticketing systems.

Audience: A number of possible audiences with different time/response characteristics interact with the network management system: human operators with different roles, authorities and competence levels; various integrated response systems such as trouble ticketing systems; workflow & process management systems. Each of these audiences require different actions suited to their view of the system and its behaviour. The system therefore needs to be able to differentiate between audiences and facilitate recommendations based on their individual time/response characteristics, the degree of competence of the individual user and the level of autonomy of the system.

Existing responses: The majority of responses are based on previously performed best-practice responses, for example many faults have detailed specific instructions which are followed to resolve the issue. However, it is important to realise that best-practice responses differ for different networks, operators and customers. For established networks this knowledge covers a large percentage of known faults and forms a body of pre-existing responses to known incident types. However, this body of knowledge of reactive actions may not be effective for pre-emptively preventing or dealing with incidents before they occur. In addition where a specific response does not exist or an incident occurs for the first small number of times old this can be characterises as a "*cold start*" problem.

Validation: When recommendations are suggested, they have to be sanity checked by the domain expert and the accuracy of the recommended actions has to be validated. The accuracy of validation has to be a learning component for future recommendations i.e. the recommender has to weigh the correctness of previous recommendations and adjust the production and ranking of future recommendations accordingly. After initial training with the domain expert the recommender systems must learn and provide recommendations to a level such that the operator reduces the degree of supervision (eventually close to zero).

3 E-STREAM COMPONENTS

E-stream is a composite system to recommend corrective solutions to network scenarios based on predictive patterns leading up to network incidences. These predictions and recommendations are built up from several *independent* components interacting in stages. Each component is capable of carrying out different functionalities:

1) Data reducer : reducing the incoming data
2) Correlator : filtering out *irrelevant* events and correlating events
3) Pattern matcher: modelling and matching patterns
4) Predictor: predicting future patterns
5) Recommender: recommending solutions

3.1 Data Reducer and Correlator

The Data Reducer and Correlator component builds up *smart reduction* mechanism to extract 'actionable insights' from network traces which are utilised by other components in the later stages. As discussed, the bursty nature and inherent variability in the trace sources further complicate the trace exploration. Data reducer mitigated the challenge of data-deluge

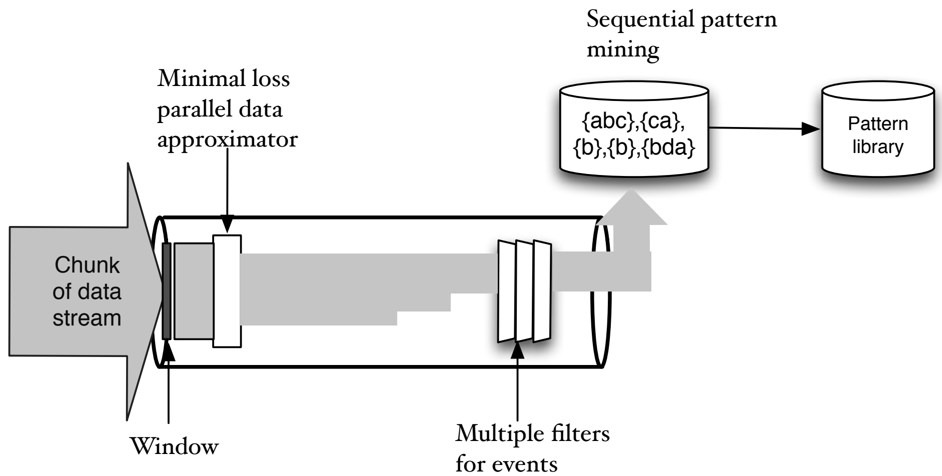


Fig. 2. The data reducer consist of the Window and Minimal Loss Parallel Data Approximation; the correlator is composed of multiple filters and sequence miner

through *controlled ingestion* of event traces into the I/O system and deploying *scaled projection* to compress the event trace information. The correlator examines the significance of events analysing the inter-event relationship through spectral distribution and temporal dependence and envelopes the relevant events traces into pseudo patterns. Accurately removing irrelevant and insignificant events from the raw data stream increases the probability of accurately finding coherent event types. Also running sequence mining techniques on reduced events is computationally much cheaper than searching for event associativity in raw data.

The functional architectures of the data reducer and correlator are entwined as shown in Figure 2. The data reducer is equipped with window based *dynamic* load shedding and Johnson-Lindenstrauss Theorem (JLT) [10] based minimal-loss approximation functionality. The correlator operates with frequency and spectral domain-based filters and with sequence mining techniques.

3.1.1 Windowing

The concept of dynamic windowing is based on *controlled* ingestion of data streams. This entails feeding the processors with manageable volumes of data while dealing with sudden and extreme influxes of event traces. The objective of the dynamic windowing process is to maintain a maximum end-to-end latency of the overall system. In this mechanism the volume of incoming data is adaptively controlled based on the data arrival rate (also defined as stream burst rate) i.e., *automatically* change the data-read rate (length of the window) based on the stream burst rate. Leveraging the data arrival rate distribution allows E-Stream to control the volume of incoming data in line with the capacity of the processors (buffer size).

3.1.2 Minimal Loss Parallel Data Approximation (MLDA)

To mitigate against very high dimensional bursty data streams, a computationally cheap dimensionality reduction technique Minimal Loss Parallel Data Approximation (MLDA) is devised. MLDA reduces event traces while efficiently approximating the degree of correlation (distance) between the events.

The basic principle of data approximation in MLDA is based on a Johnson-Lindenstrauss Theorem approach. In simple terms JLT operates according to the principle that if data points of a high dimension vector space are projected into a randomly selected subspace of with sufficiently low dimensionality, then with high probability the proportional distances between pairs of data points are preserved with a certain level of approximation. Principal Component Analysis (PCA), another state-of-the-art statistical reduction technique is more accurate in approximating the data and reducing dimensions, but PCA is computationally infeasible

$$\mathrm{A} \quad \Phi \quad \Rightarrow \quad \mathrm{Y}$$

$$\begin{array}{c|c|c|c|c}
 & e_1 & e_2 & \cdots & e_N \\ \hline
\Delta_{t_1} & & & \cdots & \\ \hline
\Delta_{t_2} & & & \cdots & \\ \hline
\vdots & & & & \vdots \\ \hline
\Delta_{t_L} & & & \cdots &
\end{array}
\quad \Phi \quad \Rightarrow \quad
\begin{array}{c|c|c|c}
 & f_1 & \cdots & f_M \\ \hline
\Delta_{t_1} & & \cdots & \\ \hline
\Delta_{t_2} & & \cdots & \\ \hline
\vdots & & & \vdots \\ \hline
\Delta_{t_L} & & \cdots &
\end{array}$$

Fig. 3. Data reduction through transformation operator Φ

for very high-dimensional data [11]. In order to reduce the distortion of JLT, a minimal loss approximation is proposed. The minimal loss criterion is based on an extension of JLT by combining a Chernoff bound: if the projection is repeated $O\left(log\frac{1}{\delta}\right)$ times, and the median of the distance between the projections is taken, the probability of accurate approximation is increased to $1-\delta$. This procedure can be represented in terms of the Law of Large Numbers (LLN), whereby performing an estimation procedure function multiple times (e.g. repeatedly measure distance between sampled and original points), this will increase the probability of accurate estimation of the function.

In summary, MLDA embeds N events recorded in L timeslots of length Δ_t into a space of lower dimension M, such that all distances are *almost* preserved through the transformation operator Φ. The scheme is shown in Figure 3, where matrix A represents the event. Iterating this procedure of approximating the distances multiple times following the principle of Law of Large Numbers can produce more condensed stream of event traces over which the correlation techniques can find pseudo patterns of events more accurately [12].

3.1.3 Online filtering

The functionality of the event filtering is akin to data reduction with the objectives to abstract low-level event information in a more structured way and to quantify the inter-relationships between network event traces. The events are reduced by finding correlated events, defining cluster event prototypes, and filtering out noise events. Isolated network events such as periodic reporting events or routine configuration events, appear in a standalone manner and are unrelated to other network events, and so show low correlations with other events around them, therefore they can be identified and treated as noise events.

Based on this objective the following online filters are deployed:

1) Spectral filter to find correlations and remove noise
2) Temporal filter to define cluster of events based on the temporal distances of co-occurrence
3) Periodicity filter to filter out periodic events

The principle of the spectral filtering technique is based on Random Matrix Theory (RMT). According to RMT a confidence band derived from the eigen-value distribution of random matrices can be utilised to separate the true “signal” from the random “noise” of a correlation matrix. The spectral filter analyses the eigen-space (spectrum) of the correlation matrix of the observed events and decompose the matrix into two parts, one part exhibiting “strong” correlative structure between the events and another part with “weak” spectral condition. The later part is treated as noise and is removed from the stream by the spectral filter. In this way the spectral filter acts as both correlation and noise filter.

The temporal distance between symptoms and effects of any network incident even spread over several different windows are approximately homogeneous. Therefore, statistically, events appearing together within a *specific time distance* can be defined as cluster of contiguous events. The temporal filter applies temporal distance based clustering to find these clusters in the correlated events.

In order to extend noise filtering with the ability to detect event patterns with periodic occurrences, a filter based on frequency domain is embedded after the temporal filtering. After noise is removed and the data density increased, the final task is to define the sequential relationships between the correlated events.

3.1.4 Sequence Miner

Correlated event traces reveal only the superficial relationship between the events. In order to extract patterns from the correlated events sequential mining techniques are applied. Association rule mining algorithms are capable of exploring the sequential relationship between the events to identify the order of occurrence and degree of association between events. This component employs association rule mining techniques over the correlated event traces to identify how event sequences are associated with actual incidents and forms an event-pattern. The magnitude of the association is quantified by several metrics. Once association rules between the events and incidents are established, these pattern rules are stored in a pattern library.

3.2 Pattern Matcher and Predictor

This component carries out two main tasks:

- encode and model patterns based on the association probability metrics drawn from the sequence miner.
- predict the occurrence of event patterns based on matching the occurrence of some of the events (the pattern 'head') in the pattern.

3.2.1 Pattern Modelling

This component uses the pattern definitions (association rules) and their associativity metrics from the sequence miner to formulate pattern models. Firstly, statistical similarity analysis is used to discover the relevance between the event sequences in the pattern definitions. The similarity between the event sequence(s) preceding a response (result or consequence) sequence provides the semantics to define a pattern model. This information about the relationships between antecedent-response sequences is important later for the recommender to assess the accuracy of the recommended actions for the predicted incidents. Hamming distance-based Locally Sensitive Hashing (LSH) is used here to compute the similarity between the pattern models.

Conventional associativity metrics, *support*, *confidence* and *lift* are computed for each event-pattern to characterize the formulated pattern model. The support counts the frequency of an event-pattern, confidence computes the probability of a specific antecedent-response forming an event-pattern, lift calculates the likelihood of co-occurrence of a specific antecedent-response pair. The knowledge provided by the metrics can be leveraged to determine the probability of a certain event sequence and incident forming a pattern model and hence accurately predict that incident whenever the event sequence is observed.

3.2.2 Pattern Recognition and Matching

Pattern definitions or rules of association between the set of event sequences of a pattern model are exported into Extensible Markup Language (XML) using Predictive Model Markup Language (PMML). These pattern definitions are then matched in the incoming stream using an off-the-shelf CEP engine Esper[1]. When an exact match occurs (i.e. 100% probability that an incident has occurred), a notification is sent to the recommender with the details of the incident.

3.2.3 Prediction

The metric values along with the antecedent-response pairs characterise each pattern model, and these are used as attributes of the models. Running a supervised learning paradigm over these attributes of the pattern models can identify appropriate 'tail' event sequences for a given sequence 'head' of a pattern model [13]. As each event in a pattern is observed in the pattern models, the supervised models should be able to predict the pattern tails with increasing probability.

3.3 From Predictions to Recommendations

When an incident occurs, or its pattern tail/response is predicted, all relevant information is forwarded to the Recommender. The Action Selector then selects a set of candidate corrective actions from the action catalogue, informed by relevant and similar antecedent-response pairs and associated human-informed actions that were previously applied. The action list is then sorted and ranked by the action

1. EsperTech: http://esper.codehaus.org

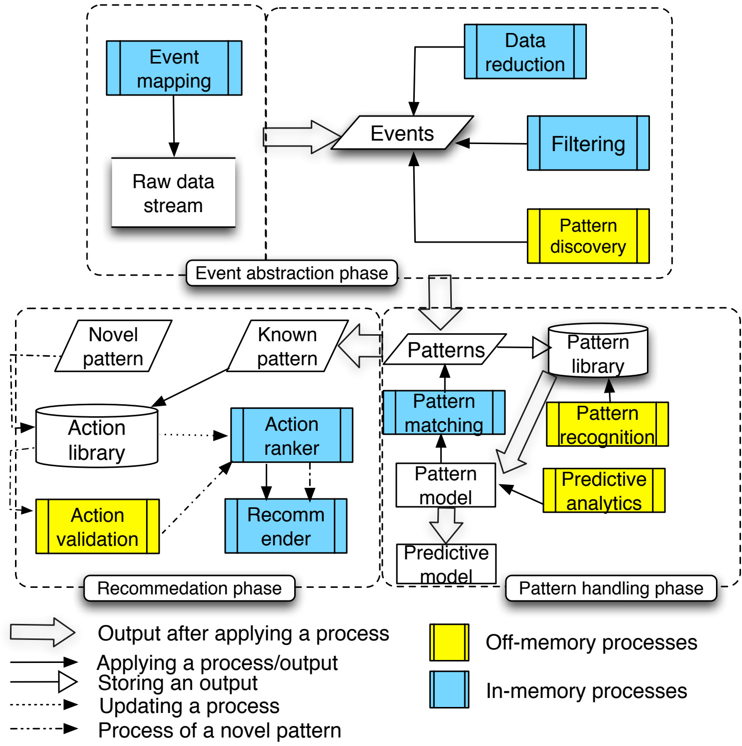


Fig. 4. Different transformation phases and computing platforms for processing the network trace information

ranker based on ranking parameters derived from guidelines of best practice and/or the historical adoption of similar actions in response to similar past incident indications. A ranked list of suggested actions may then be further manipulated, e.g. select only the highest ranked suggestions. Because more than one recommendation is presented, but the recommendations are ranked, the management agent can quickly see alternative recommendations to a given incident or incident prediction, and the degree to which those recommendations are deemed suitable. The manager then retains the authority to re-rank the list, select one or more of the recommendations, ignore all recommendations, or explicitly select, define or refine a different action. This step provides feedback to the recommendation system to adjust or extend its action catalogue and refine its ranking parameters. Using this continuous feedback process, the recommender system learns and evolves. This feedback mechanism provides a way to deal with the "cold start" problem inherent in all recommender systems. This approach also provides a mechanism to evolve as the network, context, institutional knowledge, or business priorities for the network evolves.

4 INFORMATION PROCESSING

E-stream components perform online processing of incoming information (the e-streams), transfers outputs to the next component in turn for further processing, and finally recommends the solutions to various network incidents and incident predictions. Each processing step is designed to run on appropriate computing platforms to maintain scalability of E-stream as an end-to-end system. The flow and transformation of the traces is depicted in Figure 4. In the figure, the processes running on different computing platforms are indicated with separate colours.

4.1 Information flow

Streamed trace information flows through each component and undergoes different transformation to finally trigger an action (or a set of actions). Based on the objective and functionality of the components, the transformation of the traces can be categorized into three phases.

Event abstraction phase: In this phase high volume event traces (e-streams) originating from different network management sources are compressed into an abstract form defined as patterns. At each step of this phase, the volume of data is reduced and abstracted by data reducer, filters and sequential miner components of E-Stream. Event information is extracted from the input raw data using event mapping. Events are matched with the respective sources and normalised. The data reduction procedure then synopsizes the events incurring minimal data information loss. Filtering techniques then correlate events, remove noise and cluster similar event types. The ordering of the filtering tasks is based on the context of the trace; removing noise increases the probability of accurately finding relevant event types. The final outcomes of this phase are the discovered event sequences incorporating association rules and the event relationship metrics.

Pattern handling phase: In this phase patterns stored in the pattern library are used to recognize patterns and build *pattern models*. The patterns are then matched in the incoming streams. The association rules and relationship metrics from the pattern models are then used to calculate the increasing likelihood of earlier events in a pattern being able to predict later events in each particular pattern. In the supervised learning framework, an incident is predicted whenever the probability is higher than a predefined threshold of certainty.

Recommendation phase: In the third and last phase *actions* are suggested for *known* patterns which are successfully recognized, matched and predicted in the previous phase. For matched patterns specific actions are recommended from the action library; for the predicted patterns the actions from the library are first ranked based on the action-response relationship and then the top actions are recommended. Event sequences without any prior profile (or unrecognised patterns) are defined as *novel* patterns. Actions for these patterns are first selected based on the proximity with other pattern models in the pattern library. Then these set of actions are validated in order to be able to recommend most appropriate actions.

4.2 Scalability and optimization of information processing

Online information processing has 3 limiting factors: scale, precision, and timeliness. To address scale, components are designed to exploit parallel computation and distributed processing tools. We consider two different aspects of parallel computing here: (a) pipeline parallelism and (b) partial parallelism to execute the tasks of each component. Based on the throughput, tasks are implemented on in-memory and off-memory processing platforms to facilitate restricted latencies. In Figure 4, the processes running on in-memory are coloured light-blue and the processes running on off-memory are coloured yellow.

a. in-memory processing: In E-stream, data reduction, filtering, pattern matching and action recommendation for matched patterns require the data structure to be preserved, for information to be passed quickly and easily through access points, and finally need to process the information in near real-time. In-memory processing is most suitable for these complex and time-sensitive tasks to achieve increasing speed and reliability to deliver the output within a limited time-delay.

b. off-memory processing: off-memory processing is typically disk-based, meaning the application queries data stored on off-RAM. In contrast to in-memory, off-memory processing can deal with huge amounts of data. The sequential pattern discovery task is specifically implemented to utilize disk based processing for high throughput. Due to the data intensive nature of pattern recognition tasks, pattern model building for matching patterns, and prediction of patterns, these task are required run off-memory.

Among the analytical tasks (coloured in yellow in Figure 4), data reduction and filtering are realised using the Storm[2] stream computing framework. Pattern matching is realized using the Hadoop-MapReduce framework which leverage the MapReduce computing framework for the exact matching tasks.

2. http://storm-project.net/

5 CONCLUSION

Large complex systems such as telecommunications networks are very difficult and costly to manage. Predicting, preventing, mitigating and fixing problems as early in the discovery cycle as possible is a key strategy in reducing Operational Expenditure (OPEX). The Predict & Recommend approach presented in this paper has the advantage that the network management system benefits from automated analysis and from learning human experts' approaches. Human operators should retain the ultimate decision to ignore or select from recommendations presented, until the system provides accurate and confident predictions with high-value benefits. Thus the recommender system starts as an assistant, but can later have authority to perform automated tasks delegated or revoked. This allows human operators to concentrate on high-value cases, exceptions and critical situations not yet sufficiently learned by the prediction and recommender systems.

Using prediction and recommender systems as presented in this paper will support faster resolution of network issues by presenting candidate solutions, rather than simply presenting a list of incidents. As the system learns, tuning "best practice" for a given network deployment and behaviour characteristics, the amount of mundane troubleshooting required by the NOC personnel for day-to-day operation and maintenance of the network is reduced, thus reducing cost, improving management throughput, and freeing up time and resources for the managers to concentrate on more strategic management issues.

REFERENCES

[1] Ericsson: *More than 50 billion connected devices*, http://www.ericsson.com/res/docs/whitepapers/wp-50-billions.pdf (Ericsson Whitepaper: February 2011)

[2] Infonetics Research, *Subscriber Data Management Software and Services*, 2nd Edition, Nov, 2011.

[3] B. Jennings, S. van der Meer, S. Balasubramaniam, D. Botvich, M. O Foghlu, W. Donnelly, J. Strassner, "Towards autonomic management of communications networks," *IEEE Commun. Mag.*, vol 45, no 10, pp. 112–121, 2007.

[4] R. Chaparadza, "UniFAFF: a unified framework for implementing autonomic fault management and failure detection for self-managing networks", *Int. J. Network Mgmt.*, vol. 19, no. 4, pp. 271–290, 2009.

[5] W. Zirkel, G. Wirtz, "A process for identifying predictive correlation patterns in service management systems," in *Proc. Int. Conf. on Service Systems and Service Management (ICSSSM) 2010*, pp. 1–6, 2010.

[6] J. Keeney, S. van der Meer and G. Hogan, "A Recommender-System for Telecommunicatons Network Management Actions," in *Proc. Int. Symp. on Integrated Network Management 2013–TechSessions*, IFIP/IEEE, pp. 760–763, Ghent, 2013.

[7] A. Brito, A. Martin, T. Knauth, S. Creutz, D. Becker, S. Weigert, C. Fetzer, "Scalable and Low-Latency Data Processing with Stream MapReduce," in *Proc. Int. Conf. on Cloud Computing Technology and Sci. (CloudCom)*, IEEE, pp. 48–58, 2011.

[8] T. Wang, M. Srivatsa, D. Agrawal, and L. Liu. "Spatio-temporal patterns in network events," in *Proc. Int. COnf. (Co-NEXT '10)*, ACM, New York, Article 3, 2010.

[9] X. Meng, G. Jiang, H. Zhang, H. Chen, K. Yoshihira, "Automatic Profiling of Network Event Sequences: Algorithm and Applications," in *Proc. 27th Conf. on Comput. Commun. (INFOCOM 2008)*, IEEE, pp. 266–270, 2008.

[10] W.B. Johnson, J. Lindenstrauss, "Extensions of Lipschitz mappings into a Hilbert space," in *Proc. Conf. in Modern Anal. & Probability (New Haven, Conn., 1984)*, Contemporary Mathematics 26, Providence, RI: American Mathematical Society, pp. 189206, 1984.

[11] D. Fradkin, D. Madigan, "Experiments with random projections for machine learning", in *Proc. 9th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD '03)*, ACM, New York, NY, USA, pp. 517–522, 2003.

[12] G. Reeves, J. Liu, S. Nath, and F. Zhao, "Managing massive time series streams with multi-scale compressed trickles," in *Proc. VLDB Endow.*, vol. 2, no. 1, pp. 97-108, Aug. 2009.

[13] C. Rudin, B. Letham, and D. Madigan, "Learning theory analysis for association rules and sequential event prediction," *J. Mach. Learning Research*, vol. 14, no. 1 (January 2013), pp. 3441–3492, 2013.

[14] F. Zaman, S. Robitzsch, Z. Wu; J. Keeney, S. van der Meer, G.-M. Muntean, "A heuristic correlation algorithm for data reduction through noise detection in stream-based communication management systems," in *Proc. Symp. Network Operations and Manage. (NOMS)*, IEEE, pp. 1–8, 2014.

[15] *E-Stream : Innovative Performance Management System for Heterogeneous Networks*, http://www.estream-project.com

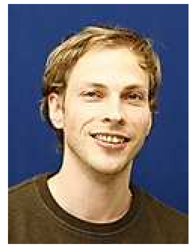
Faisal Zaman (zaman.faisal@ieee.org)is a Post doctoral researcher with the Performance Engineering Laboratory and Network Innovations Centre, Rince Institute, Dublin City University, Ireland. He received his PhD in Information Science from Kyushu Institute of Technology in 2011.

In his previous tenure as a Post Doctoral researcher in Kyushu Institute of Technology, he analysed time series data for weather forecasting and micro-array data for gene classification. He also worked as a Statistical Programmer in Shafi Consultancy Ltd and lead analytical teams to analyse medical trial data.

He is program committee member of several data mining conferences. He has published 30 articles, conference proceedings, books, book chapters, conference papers, and technical reports. He has experience in supervising PhD and M.Sc. level students.

Gabriel Hogan (ghogan@cngl.ie) is the Intellectual Property Manager at the Centre for Global Intelligent Content (CNGL). His primary focus is enabling research impact through collaborative innovation and entrepreneurship. He holds a BSc in Software and Mathematics and an MSc in Technology & Innovation Management and has over 25 years experience in ICT in the corporate, start-ups and academic sectors.

Gabriel was previously Head of Research & Innovation at LM Ericsson Ireland where he directed a number of research teams and chaired Ericssons' Network and Service Management Patent Board. He has published on and filed intellectual property across the areas of Efficient Mobile Power, Recommender Systems, Predictive Analytics, Stream Processing and Network Performance Analysis. He has co-ordinated and led a number of Industry/Business centred EU and Irish-government funded research projects.



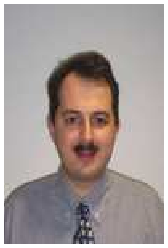
Sven van der Meer (vdmeer@ieee.com) received his PhD in 2002 from Technical University Berlin. He joined Ericsson in 2011 where he is currently a Master Engineer leading a team that will enhance the capabilities of Ericsson's OSS products. Most of his current time is dedicated to design and build advanced policy and predictive analytics systems.

In the past, Sven has worked with Fraunhofer FOKUS (Berlin, Germany), Technical University Berlin (Germany) and the Telecommunication Software and Systems Group (TSSG, Ireland), leading teams and projects, consulting partners and customers, and teaching on university level.

He is actively involved in the IEEE CNOM community as standing member of programme committees (IM, NOMS, CNMS, and APNOMS amongst others) and has helped to create and organise successful workshop serieses (MACE, MUCS, and ManFed.Com amongst others). He also contributed to standardisation organisations, namely the OMG and the TM Forum. He has published in more than 100 articles, conference proceedings, books, book chapters, conference papers, and technical reports. He has supervised and evaluated 6 PhD and more than 30 M.Sc. students.

John Keeney is a Senior Researcher in LM Ericsson Ireland, working in the Network Management laboratory in Ericsson's Software Research Campus in Athlone. His research focus is on monitoring and managing complex systems, especially telecoms systems, with a particular focus on knowledge extraction, event stream processing, and performance analysis. His work in Ericsson centres on online analytics and optimisation of Radio Access Network performance to inform the next-generation of Operation System Support (OSS) concepts for Ericsson's OSS product unit.

Sebastian Robitzsch is a Post-doctoral Researcher with the Rince research institute at Dublin City University, Ireland. He received his PhD from University College Dublin, Ireland in 2013 and an M.Sc. equivalent (Dipl.-Ing. (FH)) from University of Applied Sciences Merseburg, Germany. In the past he has worked with T-Systems (Germany), Fraunhofer FOKUS (Germany) and Nokia Research Centre (Finland). His research spans from interference issues and self-configuration techniques in 802.11-based multi- antenna mesh networks, heterogeneous radio access networks to system architecture design for trace analytics and recommender systems for Next Generation Operations Support Systems (NGOSSs). Furthermore, he is involved in ongoing European research activities focusing on the 5th generation of mobile networks.



Gabriel Miro-Muntean (gabriel.muntean@dcu.ie) received the Ph.D. degree from Dublin City University, Dublin, Ireland, for research in the area of quality-oriented adaptive multimedia streaming in 2003. He is a Senior Lecturer with the School of Electronic Engineering at Dublin City University (DCU), Dublin, Ireland. He is a Co-Director of the DCU Performance Engineering Laboratory and Consultant Professor with Beijing University of Posts and Telecommunications, China.

His research interests include quality-oriented and performance-related issues of adaptive multimedia delivery, performance of wired and wireless communications, energy-aware networking, and personalized e-learning. He has published over 180 papers in prestigious international journals and conferences, has authored three books and 15 book chapters and has edited six other books. He is an Associate Editor of the IEEE Transactions on Broadcasting, Associate Editor of the IEEE Communications Surveys and Tutorials, and reviewer for other important international journals, conferences, and funding agencies. He is a member of ACM, IEEE and IEEE Broadcast Technology society.

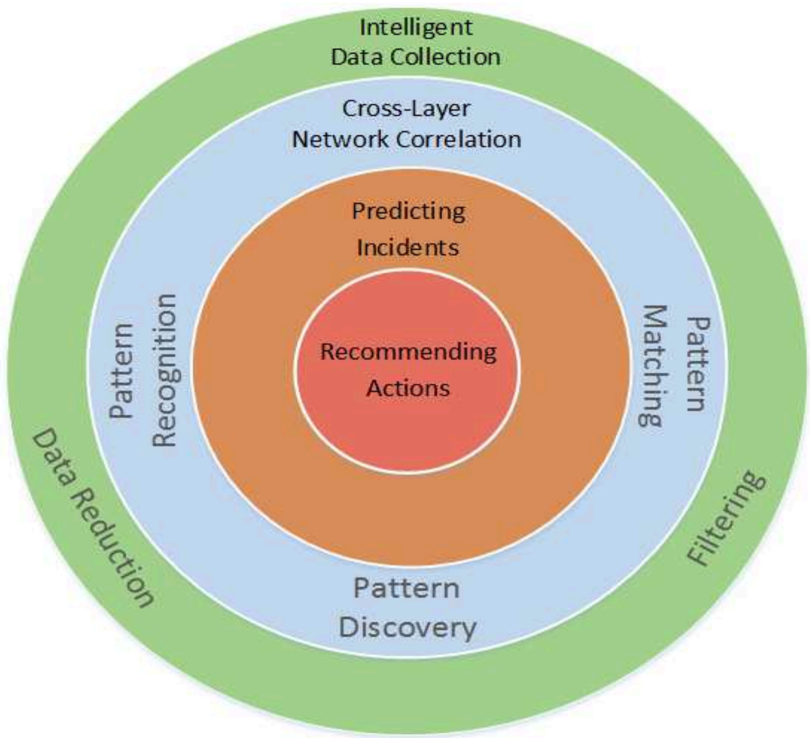


Fig. 5. E-Stream components in different information processing layers.

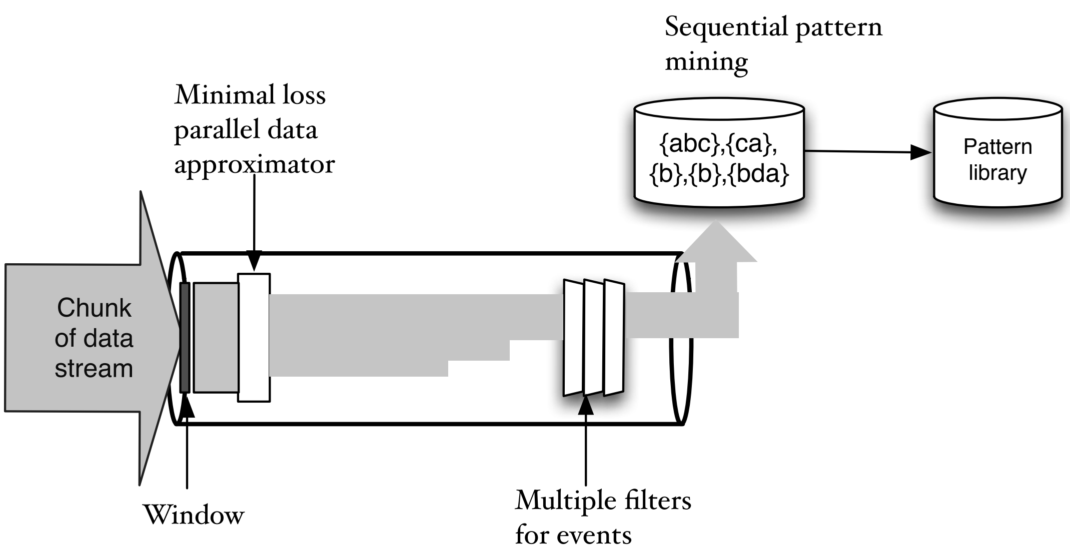


Fig. 6. The data reducer consist of the Window and Minimal Loss Parallel Data Approximation; the correlator is composed of multiple filters and sequence miner

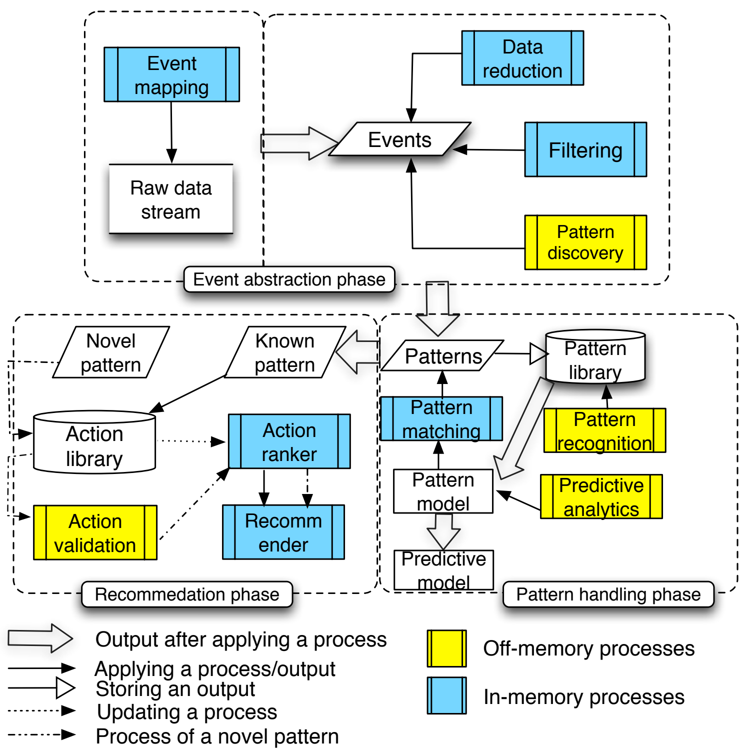


Fig. 7. Different transformation phases and computing platforms for processing the network trace information