

Améliorer la découverte de chroniques par une découpe intelligente d'un log d'alarmes

Françoise Fessant, Christophe Dousson, Fabrice Clérot

France Télécom R&D, 2 avenue P. Marzin, 22307 Lannion
{francoise.fessant, christophe.dousson, fabrice.clerot}@francetelecom.com
<http://www.rd.francetelecom.com>

Résumé. Cet article décrit une méthode de prétraitement destinée à faciliter la découverte de motifs fréquents dans un log d'alarmes. Au cours d'une première étape les types d'alarmes qui présentent un comportement temporel similaire sont regroupés à l'aide d'une carte auto-organisatrice. Puis on recherche les parties du log qui sont riches en alarmes pour les différents groupes. Des sous logs sont construits à partir des alarmes des zones sélectionnées. La méthode a été validée sur un log provenant d'un réseau ATM.

1 Introduction

La complexité croissante des réseaux de télécommunications nécessite le développement d'outils de supervision et de corrélation d'alarmes pour aider les opérateurs à contrôler leurs réseaux. Ces outils sont chargés de diminuer la quantité d'informations remontée et de focaliser l'attention de l'opérateur sur des problèmes critiques. Le principal obstacle à la mise en œuvre de ces outils est la difficulté d'acquisition de l'expertise nécessaire à leur fonctionnement (le plus souvent sous forme de règles, Moller et al. 1995, Nygate 1995).

Le logiciel FACE (Frequency Analyser for Chronicle Extraction) apporte une aide à l'acquisition d'expertise en analysant les journaux d'alarmes. A partir de la fréquence d'apparition des alarmes, FACE va découvrir et construire automatiquement plusieurs motifs temporels plus ou moins complexes qui se sont produits un certain nombre de fois dans le log (ces motifs sont baptisés « modèles de chronique », Dousson et al. 1999). FACE peut aider à réduire le flot d'alarmes présenté à l'opérateur lors de la supervision : si une chronique correspond à un défaut de fonctionnement elle sera remontée à l'opérateur, sinon celle-ci pourra être filtrée. La qualification défaut/normal d'une chronique reste du ressort d'un expert chargé d'analyser les modèles de chroniques découverts par les algorithmes.

Le processus de recherche de chroniques implémenté dans FACE repose sur une exploration exhaustive des instances des chroniques dans le log et est donc très consommateur d'espace mémoire. Le facteur principal responsable de cette explosion est la taille du log.

Actuellement les utilisateurs de FACE, pour s'affranchir de ce problème, sont amenés à sélectionner certains types d'alarmes et/ou certaines périodes temporelles dans le log de manière à extraire des morceaux du log qui puissent être traités par l'outil. Le but du travail décrit ici est de permettre l'extraction automatique de morceaux du log pertinents de façon à s'affranchir de l'étape manuelle de prétraitement.

2 Description et représentation des données

Les instances des chroniques ne s'accumulent pas de manière aléatoire dans le temps mais ont tendance à se grouper sur des périodes de temps limitées. Ce phénomène peut s'expliquer par la nature des alarmes qui se produisent essentiellement quand le réseau rencontre une configuration spécifique. A cet instant, qui peut représenter une faible partie du journal, on a des chances de trouver des chroniques relatives à cette configuration qui ne se reproduira peut-être pas dans le reste du journal.

La méthode de prétraitement exploite cette propriété et cherche à regrouper dans des sous journaux des alarmes qui présentent un comportement temporel similaire.

Dans une première phase de préparation, le journal des alarmes est découpé en tranches qui contiennent un nombre variable d'alarmes, certaines pouvant même être vides. Ces tranches forment une partition du journal et constituent les unités de base sur lesquelles vont être décrits les types d'alarmes et à partir desquels les sous journaux vont être construits.

Chaque type d'alarme peut maintenant être décrit sur l'espace des tranches. Un profil d'alarme est donné par le compte du nombre des occurrences par tranche. Le profil cumulé de l'alarme est obtenu ensuite en sommant les valeurs des occurrences des périodes successives. Les valeurs sont normalisées de manière à ce que la somme des occurrences du type d'alarme soit égale à un ; de cette manière on rend les profils cumulés indépendants du nombre total d'occurrences. Avec cette description, les alarmes sont représentées d'une manière qui prend en compte leur nature temporelle.

La méthode a été mise en oeuvre sur un log d'alarmes réel provenant d'un réseau ATM d'une durée d'un mois, avec environ 46000 lignes d'alarmes (réparties en 12160 types différents). La taille de la tranche a été fixée à 15 minutes ; on a obtenu 2317 tranches de 16 alarmes en moyenne.

Le but de l'étape suivante est de regrouper les types d'alarmes qui présentent un comportement temporel similaire.

3 Segmentation des profils cumulés

Les types d'alarmes sont regroupés à l'aide d'une carte auto-organisatrice (SOM, Kohonen 2001) à laquelle on applique, après auto-organisation, une procédure de classification hiérarchique ascendante qui facilite l'analyse quantitative de la carte (Vesanto et al. 2000).

On considère uniquement les types d'alarmes fréquents (qui totalisent au moins trois occurrences dans le log). Au final, l'analyse est effectuée à partir d'une base de 3042 profils cumulés décrits sur l'espace de 2317 tranches (75% des types d'alarmes pèsent moins de trois occurrences).

La carte après auto-organisation a produit 10 groupes. Chaque groupe est caractérisé par un profil moyen calculé par la moyenne des profils cumulés qui ont été rangés dans le groupe (figure 1). Chaque groupe est identifié sur la figure par un numéro, on donne les tranches en abscisse et la valeur du profil moyen du groupe en ordonnée.

On observe des comportements d'accumulation très différents : les groupes 1, 8 et 9 regroupent des profils cumulés d'alarmes qui se produisent dans une zone temporelle réduite du log (les alarmes du groupe 8 s'accumulent au début du log, celles du groupe 9 s'accumulent à la fin et celles du groupe 1 s'accumulent au milieu du log). Les profils d'accumulation de ces groupes ont une pente très forte, traduisant une accumulation très

rapide des alarmes. Les alarmes, dont les profils cumulés sont regroupés par les groupes 3 ou 4 par exemple, se produisent tout au long du log et les profils d'accumulation correspondant présentent des profils plus doux que ceux des autres clusters.

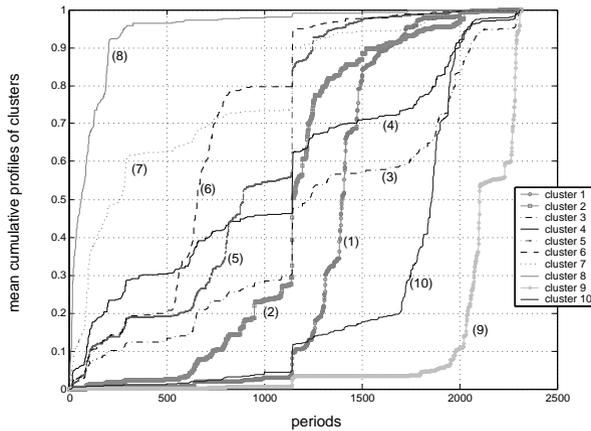


FIG. 1 – *profils cumulés moyens des groupes.*

A la fin de cette étape, on dispose d'un nombre limité de groupes qui résument et caractérisent l'ensemble des types d'alarmes : tous les types d'alarmes classifiés dans un groupe s'accroissent de la même manière à travers le log.

4 Construction des sous logs

Certaines tranches du log sont plus importantes que d'autres pour un groupe donné de types d'alarmes. On définit l'importance d'une tranche pour un groupe par le nombre d'alarmes du groupe qu'elle contient proportionnellement au nombre total de ses alarmes. Les tranches les plus importantes pour le groupe sont ensuite sélectionnées de la manière suivante : les périodes sont triées en fonction de leur importance pour le groupe ; les périodes qui présentent l'importance la plus élevée sont rangées en premier. Puis on sélectionne les périodes jusqu'à obtenir $x\%$ de la distribution des alarmes du groupe. On fixe comme critère d'arrêt $x=90\%$ (on retient les tranches jusqu'à ce qu'on obtienne 90% de la distribution des alarmes du groupe).

On résume sur la figure 2 la localisation des périodes sélectionnées pour chaque groupe. Les périodes coïncident très clairement avec les zones d'accumulation du profil moyen du groupe données figure 1.

Le nombre de périodes à prendre en compte pour obtenir 90% des alarmes d'un groupe est très variable selon les groupes. On sélectionne peu de périodes pour les alarmes qui se produisent dans une zone de temps réduite et beaucoup de périodes pour les alarmes qui se produisent tout au long du log. Cependant, quel que soit le groupe et le nombre de périodes retenues, l'ensemble des alarmes correspondant ne dépasse pas 35% de la totalité des alarmes du log.

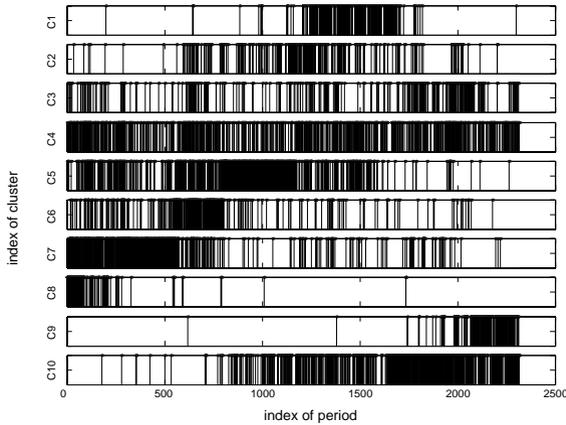


FIG. 2 – localisation temporelle des tranches sélectionnées pour les 10 groupes.

On dispose maintenant d'autant d'ensembles de périodes que de groupes. Ces ensembles de périodes vont nous servir à construire les sous logs.

Un sous log est attaché à chaque groupe de profils cumulés d'alarmes. On construit le sous log simplement en listant et en réordonnant temporellement les alarmes contenues dans les périodes retenues pour le groupe correspondant. Toutes les alarmes sont prises en compte, y compris les alarmes de type peu fréquent.

La méthode de sélection des tranches décrite ci-dessus considère chaque groupe de manière indépendante, aussi, une même tranche peut être attribuée à plusieurs groupes. En conséquence, l'ensemble des alarmes contenues dans les groupes totalise 2,5 fois le nombre des alarmes du log initial. Au final, on a créé 10 sous logs à partir du log initial. La recherche de chroniques peut ensuite être effectuée dans les sous logs, chacun d'entre eux étant traité séparément.

5 Résultats expérimentaux

Ce paragraphe résume les résultats expérimentaux obtenus avec FACE lors du traitement du log entier et des sous logs construits à l'aide de la méthode de prétraitement (sous logs SOM).

La mise en œuvre de l'algorithme d'apprentissage de FACE nécessite le réglage de deux paramètres : la fenêtre temporelle t_w qui fixe la durée maximale qui doit exister entre les alarmes d'un modèle de chronique et le nombre minimum d'instances n_{qmin} que le modèle de chronique doit avoir dans le log pour être considéré comme fréquent. La valeur de t_w est fixée à 15 secondes et gardée constante ; on choisit de faire varier la valeur de n_{qmin} .

La figure 3 illustre les principaux résultats obtenus. On donne le nombre de modèles de chroniques différents découverts à partir du log entier et à partir des sous logs SOM, en fonction de n_{qmin} . On indique également ce nombre pour des sous logs construits manuellement - comme pourrait le faire un opérateur - en découpant le log en plusieurs

tranches successives contenant chacune un même nombre d'alarmes. Quatre sous ensembles de sous logs ont été testés (avec un découpage en 2, 3, 5 et 10 sous logs).

Toutes les expérimentations ont été effectuées à partir du même PC (en l'occurrence un Pentium 4 avec 1,7 GHz de CPU et 1 Go de RAM), FACE étant la seule application à fonctionner sur le PC.

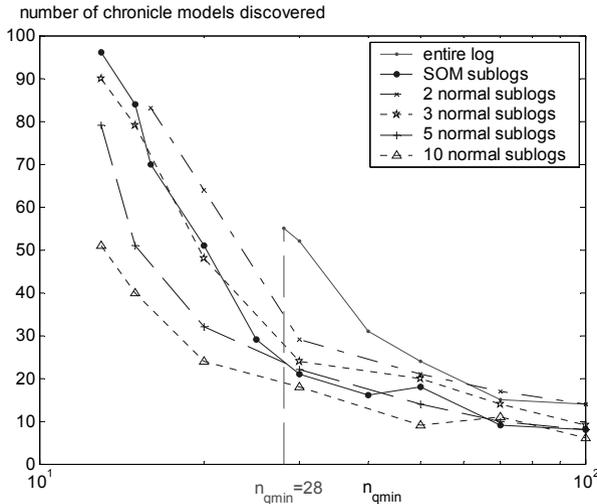


FIG. 3 – nombre des modèles de chroniques découverts.

Les différentes courbes montrent que le log entier ne peut pas être traité pour des valeurs de n_{qmin} inférieures à 28 ; la raison étant une saturation de l'espace mémoire du calculateur. Le traitement reste possible pour les sous-logs pour des valeurs inférieures de n_{qmin} (la limite est atteinte pour $n_{qmin} = 16$ pour l'ensemble des 2 sous logs et pour $n_{qmin} = 13$ pour les autres ensembles de sous logs, et il n'a pas été possible d'atteindre une valeur plus basse).

On découvre au final plus de chroniques avec les sous logs qu'avec le log entier (sauf pour l'ensemble des 10 sous logs) : pour une valeur donnée de n_{qmin} , on découvre plus de chroniques avec le log entier qu'avec les sous logs, mais l'exploration peut être poussée sur les sous logs pour de plus faibles valeurs de n_{qmin} . De plus, une analyse détaillée des modèles de chroniques obtenus montre que l'on retrouve dans les sous logs tous les modèles de chroniques découverts à partir du log entier, ainsi que de nouvelles chroniques.

Si on s'intéresse maintenant au nombre de modèles de chroniques découverts pour les plus faibles valeurs de n_{qmin} sur les différents types de sous logs, on constate que les meilleurs résultats sont obtenus pour les sous logs SOM. L'amélioration obtenue n'est pas due au découpage du log en sous logs, ni à la capacité d'atteindre des faibles valeurs de fréquences. L'amélioration doit être attribuée à la méthode de prétraitement adoptée pour la construction des sous logs.

On notera de plus que la recherche manuelle du bon nombre de sous logs, qui permet de découvrir le maximum de modèles de chroniques, est un processus long à mettre en œuvre. La méthode de prétraitement que nous proposons extrait automatiquement les sous logs pertinents.

Améliorer la découverte de chroniques par une découpe intelligente d'un log d'alarmes

Ces résultats expérimentaux valident la méthode de prétraitement qui s'est révélée très efficace : on a retrouvé dans les sous logs tous les modèles de chroniques qui avaient pu être extraits du log entier, ainsi que de nombreux nouveaux modèles. De plus, la durée totale du traitement des sous logs est du même ordre de grandeur que celle du traitement du log entier.

Pour plus de précisions sur le processus de prétraitement du log se reporter à (Fessant et al. 2004).

Références

- Dousson C. et Vu Duong T. (1999), Discovering chronicles with numerical time constraints from alarm logs for monitoring dynamic systems, Proceedings of the 16th IJCAI 1999, pp. 620-626
- Fessant F., Clérot F. et Dousson C. (2004), Mining of an alarm log to improve the discovery of frequent patterns, Proceedings of ICDM 2004
- Kohonen T. (2001), Self organizing maps, Springer-Verlag, 2001
- Moller M., Tretter, S. et Fink B (1995), Intelligent filtering in network-managements systems, Proceedings of the 4th ISINM 1995, pp. 304-315
- Nygate YA. (1995), Event correlation using rule and object base techniques, Proceedings of the 4th ISINM 1995, pp. 279-289
- Vesanto J. et Alhoniemi, E. (2000), Clustering of the Self Organizing Map, IEEE Transactions on Neural Networks 11 3 2000, pp. 586-600

Summary

In this paper we propose a method to pre-process a telecommunication alarm log with the aim of discovering more accurately frequent patterns. In a first step, the alarm types which present the same temporal behaviour are clustered with a self organizing map. Then the log areas which are rich in alarms of the clusters are searched. The sublogs are built based on the selected areas. We will show the efficiency of our preprocessing method through experiments on an actual alarm log from an ATM network.