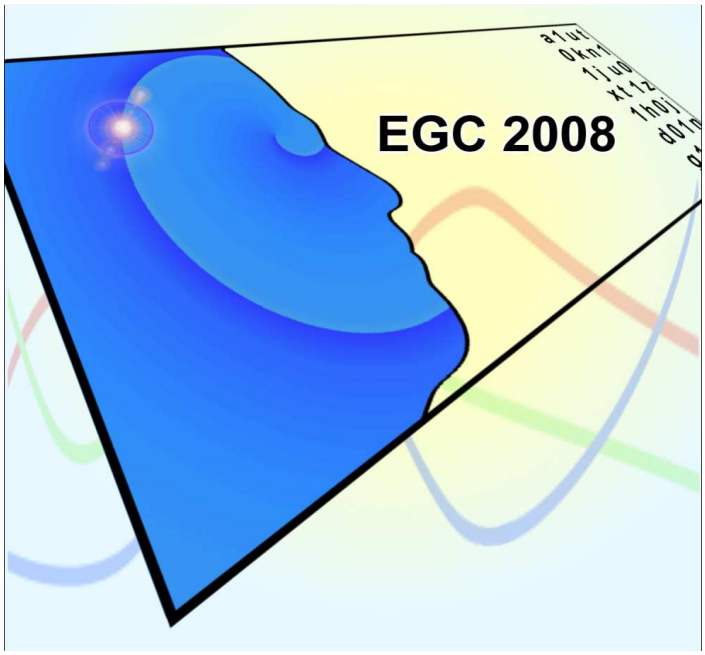
8èmes Journées Francophones
Extraction et Gestion des Connaissances
Sophia Antipolis 29 janvier 2008

Atelier



Fouille de données temporelles

Organisateurs :

- Georges Hébrail (ENST Paris)
- Pascal Poncelet (Ecole des Mines d'Alès)
- René Quiniou (IRISA/INRIA Rennes)

Responsables des Ateliers EGC :

Alzennyr Da Silva (INRIA, Rocquencourt)
Alice Marascu (INRIA, Sophia Antipolis)
Florent Masseglia (INRIA, Sophia Antipolis)

http://www-sop.inria.fr/axis/egc08

EGC

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE
INRIA
centre de recherche SOPHIA ANTIPOLIS - MÉDITERRANÉE

EGC 2008 - Atelier Fouille de Données Temporelles

Objectifs

Dans de nombreux domaines applicatifs, tels la biologie, la santé, les télécommunications, la vidéo-surveillance, l'énergie, l'environnement, etc., des données sont enregistrées de manière continue. Les bases de données concernées peuvent atteindre des tailles gigantesques ou ne retenir que les données les plus récentes. Les données représentent, par exemple, les valeurs prises par des variables mesurées à intervalles réguliers ou des événements se produisant de manière irrégulière. Dans la plupart des cas, les données présentent un caractère temporel qu'il est intéressant de caractériser : relations entre les tendances de plusieurs variables, relations temporelles entre occurrences de certains types d'événements, etc. L'exploitation de cette dimension temporelle introduit une complexité supplémentaire dans les tâches de fouille de données et d'extraction de connaissances. Ainsi, il faut tenir compte :

- des aspects métriques ou symboliques des relations temporelles traitées,
- de l'irrégularité ou du manque de synchronisation des mesures,
- du volume des données à traiter,
- de la fugacité des données et de la nécessité d'un traitement en temps-réel,
- de la nécessité/possibilité ou non d'un encodage explicite des relations temporelles des données avant leur exploitation,
- de la granularité temporelle et du caractère hétérogène des types des données pouvant avoir un impact sur les motifs susceptibles d'être découverts,
- de la nature et de l'utilisation des connaissances extraites,
- de la possibilité de prendre en compte la connaissance générale sur le domaine.

Les approches généralement suivies consistent, soit à étendre les approches classiques de la fouille de données pour prendre en compte la dimension temporelle, soit à proposer de nouvelles solutions et algorithmes appropriés aux données temporelles. Dans les deux cas, elles doivent tenir compte de la complexité des algorithmes utilisés et de leur possibilité de "passer à l'échelle". L'objectif de cet atelier est de rassembler des chercheurs, du domaine académique ou de l'industrie, travaillant sur des problèmes cités ci-dessus ou sur des applications confrontées à ces problèmes.

Comité scientifique

TABLE DES MATIERES DES COMMUNICATIONS

Etude préliminaire à un modèle de prévision à court terme de l'activité d'un transporteur sous température dirigée

Wilfried Despagne*

*Laboratoire SABRES,
Agrostar (le pôle systèmes d'information du Groupe STEF-TFE)
Wilfried.Despagne@stef-tfe.com

Résumé. Cet article décrit une problématique de recherche opérationnelle. Un transporteur sous température dirigée cherche à optimiser la planification de ses ressources humaines et matérielles à travers la prévision à très court terme de son activité. Le challenge réside dans le fait de trouver un modèle de prévision unique s'adaptant, sans intervention humaine, aux spécificités des 57 agences du transporteur. La matière première est l'information récoltée par le transporteur depuis plus de six ans. Les outils sont des algorithmes mathématiques utilisés pour la prévision des séries temporelles. Le travail décrit ici vise à combiner ces outils pour qu'ils extraient le maximum d'information déterministe capable d'être anticipée. L'introduction pose la problématique et son contexte économique. Elle est suivie d'un descriptif des procédures utilisées et d'un argumentaire pour défendre leur choix. Les solutions informatiques adoptées sont inventoriées. Enfin, la conclusion renvoie à des pistes d'études.

1 Introduction

La présentation suivante est le préliminaire d'un travail de recherche dans le cadre d'une thèse Cifre. Les structures dans lesquelles ce travail est mené, sont le pôle systèmes d'information du groupe STEF-TFE appelé Agrostar et le laboratoire SABRES de l'Université Bretagne Sud. Nous présentons un état des lieux de la prévision d'activité chez un transporteur sous température dirigée. Cette démarche a permis d'acquérir un point de vue formalisé du problème et une automatisation des procédures.

Le transport sous température dirigée est l'activité qui consiste à transporter de la marchandise soumise à une température définie entre -25 ˚C et +15 ˚C. Les marchandises sont essentiellement des denrées alimentaires périssables, les produits carnés, les produits de la mer, les fruits et légumes, les produits laitiers, les surgelés, mais aussi des plantes ou médicaments. Elles ont en commun d'être soumises à un cahier des charges très strict qui définit la « chaîne du froid ». La chaîne du froid est le processus qui permet de maintenir un produit à basse température. Le froid ralentit la propagation des microorganismes. La loi impose des règles en cette matière. Les arrêtés du 9 mai 1955 (réglementant l'hygiène des aliments remis directement au consommateur) et du 20 juillet 1998 (fixant les conditions techniques et

hygiéniques applicables au transport des aliments) soumettent les industriels à une obligation de résultat. À la contrainte du maintien de la température liée à la chaîne du froid, s'ajoutent celles liées au métier du transporteur. Là encore, le législateur réglemente la durée du travail d'un chauffeur routier et l'autorisation de circulation. Ce ne sont ici que des exemples de lois parmi d'autres permettant de percevoir l'ampleur des contraintes législatives en matière de transport et plus encore de transport frigorifique. Sans une maîtrise de gestion des ressources matérielles et humaines, ces contraintes font déborder les coûts de fonctionnement.

Le domaine de l'alimentaire met en relation producteurs, industriels, distributeurs et transporteurs. Ils forment une chaîne appelée « Supply Chain » (chaîne logistique, cf Ayadi (2005)). La contrainte consistant à garder la fraîcheur des produits entre leur lieu de fabrication et celui de distribution, impose aux différents maillons de la chaîne de travailler en flux tendus. La Date Limite de Consommation (DLC) est de quelques jours pour des découpes de volaille ou de 21 jours pour des yaourts. Elle gouverne la « Supply Chain » des produits frais et imprime un rythme rapide à la chaîne de distribution. Dans son mémoire, Charlotte Terrolle (2004) souligne que les industriels répondent aux commandes quotidiennes des GMS (Grandes et Moyennes Surfaces) pour approvisionner leurs entrepôts. Les quantités de ravitaillements fluctuent quotidiennement selon les sorties de caisse des GMS. Dans le cas de Carrefour, tous les soirs, toutes les données de vente en magasin sont centralisées à l'entrepôt, les commandes sont expédiées dès le lendemain. Ce réapprovisionnement automatique représente : 80% des ventes pour l'épicerie et 98% des produits frais. Pour soutenir ce rythme et alimenter les linéaires sans pour autant avoir de stocks, les acteurs doivent prévoir leurs ventes. La logique voudrait que les GMS partagent leurs prévisions de ventes, aussi bien à un rythme mensuel que quotidien, avec les industriels de l'agroalimentaire et que ces derniers les relient aux transporteurs. Mais la loi de la concurrence ne permet pas cette pratique. Les GMS craignent de donner trop d'informations aux industriels qui, se voyant dans une position de force, pourraient en profiter pour augmenter leurs prix. Ainsi, les acteurs de la chaîne logistique établissent des prévisions chacun de leur côté.

Dans le cadre de ce travail, nous proposons un système de prévision des ventes du transporteur sous température dirigée TFE. Il est la branche transport du groupe STEF-TFE, leader français du secteur de la logistique du froid. TFE dispose d'un réseau 57 agences en Europe. Une agence comporte une plate-forme (ou quai) sur laquelle est réceptionnée la marchandise. Ces quais permettent au transporteur de préparer les commandes, les trier et les étiqueter pour les dépêcher à travers l'Hexagone en moins de 48h. Dans le jargon métier c'est un transport en A pour B. Grossièrement, il existe une agence par région. D'une part, elle est en charge d'enlever la marchandise chez ses clients régionaux pour l'injecter dans le réseau qui s'occupe de la livrer à bon port. De même, elle reçoit de la marchandise du réseau et la livre aux points de vente de sa région.

Comme les autres maillons de la « Supply Chain » alimentaire, TFE travaille suivant la méthode JAT (Juste A Temps). Dans plus de 90% des cas, l'agence reçoit du client expéditeur ses ordres de transports moins de 3h avant l'enlèvement de la marchandise. Pour résumer en quelques mots la problématique, les ordres de transports tombent par fax ou EDI (Échange de Données Informatisé) à 7h du matin pour un départ entre 8h et 12H en fonction de la destination. La livraison doit se faire entre 18h et 22h le soir sur toute la France. À cause des opérations spéciales (promotions), il arrive que les quantités à transporter varient de 1 à 10 d'un jour à l'autre. Dans ces cas, comment faire face si les clients n'en communiquent pas la

période ?

Le transporteur doit donc s'adapter aux exigences de ses clients. Pour l'aider, il souhaite mettre en place un système de prévision pour l'aide à la gestion des ressources. Le système doit permettre d'anticiper le poids des marchandises à transporter et le nombre de lettres de transports à couvrir. Ces deux informations, prévues à un horizon de 15 jours et à périodicité journalière, lui permettent d'anticiper l'effectif humain et matériel sur le quai ainsi que le nombre de semi-remorques à mettre à disposition. Par souci de simplification, nous appellerons prévision de l'activité, l'anticipation du poids et du nombre de lettres de transports. Dès lors, les prévisions de l'activité visent à fournir des éléments cruciaux pour :

- la planification des ressources matérielles et humaines,
- optimiser les règles de ramasse , d'expéditions et de distribution ,
- formaliser le comportement à court, moyen et long terme des clients,
- atteindre un niveau de service élevé,
- limiter la dépendance vis-à-vis de l'incertitude.

TFE étant un groupe et ayant comme souci d'homogénéiser les procédures de traitement, il convoite un système de prévision capable de s'adapter aux spécificités des différentes agences. Les prévisions doivent être facilement consultables, conviviales et accessibles par une interface web sur l'intranet. Enfin, les objectifs de la direction sont d'atteindre une erreur de prévision quotidienne inférieure à 5%.

Générer des prévisions à partir de modèles mathématiques ne se fait pas sans un historique. Début des années 2000, le groupe STEF-TFE investit massivement en moyens informatiques. Pour répondre aux exigences de traçabilités imposées par l'Europe (texte CE n^o 178/2002), améliorer la rentabilité et du même coup le service rendu au client, le groupe utilise un Data Warehouse. C'est un entrepôt de données Oracle© dans lequel sont stockées toutes les informations relatives aux colis transportés et entreposés par le groupe. Les données proviennent d'un système d'information appelé GTI (Gestion Intégrée du Transport). Il regroupe les applications informatiques du groupe et alimente la base de données, soit plusieurs giga-octets d'information. L'information de référence est l'ordre de transport. Il donne accès aux informations de chargement et déchargement : produit, unité logistique, lieu de déchargement-chargement, date, heure, tiers expéditeur, tiers destinataire, nombre de colis, type d'emballage, poids, catégories d'emballages ... La base de données est mise à jour quotidiennement avec les données de la veille.

La suite de cet article propose une méthode de prévision qui combine méthodes endogènes et exogènes. La méthode cherche à anticiper, à un rythme quotidien et à un horizon de 15 jours, les valeurs de 3 chroniques dont la somme de deux d'entre elles est égale à la troisième. Pour retrouver une cohérence, les valeurs prévues indépendamment, sont redressées du haut vers le bas (« top-down aggregation »).

2 Le Modèle

2.1 Vue d'ensemble

Nous cherchons à modéliser trois séries temporelles. Elles reflètent les flux des marchandises sorties d'un quai. Les marchandises sont soit transportées vers un autre quai, c'est alors de l'expédition, soit livrées chez le destinataire final, c'est de la distribution. La somme des

deux donne le total des denrées traitées à quai. Les deux quantités à prévoir, le nombre de lettres de transport et le poids des marchandises associées, sont relevés quotidiennement par TFE. La taille de l'historique est de cinq ans d'observations.

Le modèle statistique choisi cherche à se rapprocher le plus possible de l'activité du transporteur en la décomposant. Supposons que l'activité d'une agence TFE est déterminée par trois composantes, des facteurs déterministes (saisonnalité, jours fériés, promotions) dont les valeurs futures sont connues et des facteurs stochastiques. Ces dernières se décomposent en variables observées (grèves, perte ou gain d'un portefeuille client) dont les valeurs futures sont inconnues et en variables non observées (liquidation judiciaire du principal concurrent) dont les valeurs passées, présentes et à venir sont inconnues. Le modèle proposé tente d'extraire de la chronique les facteurs déterministes avant d'appliquer un modèle « autoprojectif » pour estimer les variables stochastiques observées. La différence entre les résultats obtenus par le modèle et les observations résulte des variables non observées ou mal estimées.

Soit le triplet $(X_t, Y_t, Z_t) \in \mathbb{R}^3$ représente les valeurs à t des chroniques en « expédition », « distribution » et « total ». Elles sont soumises à la contrainte $X + Y = Z$. Ces valeurs sont calculées à partir d'autres chroniques ($U_{X,t}$, $U_{Y,t}$, $U_{Z,t}$) par $X_t = \omega_1 U_{X,t}$, $Y_t = \omega_2 U_{Y,t}$ et $Z_t = U_{Z,t}$. ω_1 et ω_2 sont déterminés afin de satisfaire la contrainte. Chacune des chroniques ($U_{X,t}$, $U_{Y,t}$, $U_{Z,t}$) est modélisée par les équations (1) et (2) pour être combinée dans l'équation (3).

$$U_t^1 = T_t S_t^1 (\beta^1 F)_t V_t^1 \epsilon_t^1 \quad (1)$$

$$U_t^2 = T_t S_t^2 (\beta^2 F)_t V_t^2 \epsilon_t^2 \quad (2)$$

$$U_t = \lambda U_t^1 + (1-\lambda) U_t^2 \quad (3)$$

avec T la tendance, F un vecteur binaire correspondant à des évènements calendaires, β leurs pondérations, S les coefficients saisonniers, V les processus stationnaires, ϵ les bruits blancs. Pour étudier les composantes indépendamment les unes des autres, nous utilisons la fonction logarithme népérien.

$$\begin{aligned} ln(U_t) = \lambda \left[ln(T_t) + ln(S_t^1) + ln((\beta^1 F)_t) + ln(V_t^1) + ln(\epsilon_t^1) \right] + \\ (1-\lambda) \left[ln(T_t) + ln(S_t^2) + ln((\beta^2 F)_t) + ln(V_t^2) + ln(\epsilon_t^2) \right] \end{aligned}$$

2.2 Modélisation des éléments déterministes

2.2.1 La tendance

L'activité des agences TFE connaît deux tendances, une tendance intra-annuelle et une tendance inter-annuelle. La tendance intra-annuelle décrit l'activité d'une agence entre janvier et décembre. La tendance intra-annuelle est celle à long terme. Après les fortes dépenses de fin d'année, les ménages se remettent à économiser en janvier. C'est pourquoi l'activité est forte en décembre et chute en janvier. D'autre part, dans le milieu économique, les tendances sont lentes et progressives (Burtschy (1980)). Ces constations nous font choisir une la tendance linéaire par morceau de périodicité annuelle. Son équation s'écrit :

$$T_t = \theta t + \varphi An(t) + cste$$

avec $An(t)$ l'année correspondante à la date t ; θt représente la tendance intra-annuelle, $\varphi An(t)$ la tendance inter-annuelle.

2.2.2 La saisonnalité

L'activité des agences TFE est une superposition de mouvements oscillatoires de périodes hebdomadaires et journalières. Ainsi, elle admet une double saisonnalité que nous allons estimer pour l'effacer de la chronique. La saisonnalité hebdomadaire comporte 53 coefficients et la saisonnalité journalière en comporte 318 (53 semaines $\times$ 6 jours). La saisonnalité hebdomadaire est due à des périodes d'activités fluctuantes. Elles sont causées par des évènements extérieurs tels que la météo, les vacances scolaires, les périodes de fête. Inclus dans la saisonnalité, il ne sera plus nécessaire de les analyser individuellement. L'idéal serait de s'affranchir du compteur des semaines pour ne retenir que les distances par rapport à des évènements calendaires. Ainsi, nous n'aurions plus à nous préoccuper du fait que pendant l'année A tel férié est tombé la semaine $s13$ alors que l'année suivante il est tombé la semaine $s14$. C'est une perspective à approfondir. Pour l'instant, nous retenons 53 semaines dont les deux extrêmes sont corrigés suivant le nombre de jours qu'elles comportent.
La saisonnalité journalière, très marquée, est due à une répartition de l'activité sur les 6 jours ouvrés de la semaine. Cette répartition dépend de l'agence en question. Le samedi par exemple, l'activité est réduite à son strict minimum, elle varie ensuite suivant les jours d'approvisionnements des grossistes et des GMS. Rappelons qu'ils représentent 80% des volumes transportés. Il existe de nombreuses méthodes de désaisonnalisation. Elles ont l'avantage de décrire l'activité aux décideurs. Les coefficients saisonniers montrent l'écart de la valeur moyenne constatée pour une semaine i et un jour j par rapport à la tendance. En matière de prévision des ventes, la méthode idéale n'existe pas. Partant du principe que deux valent mieux qu'une, pourquoi ne pas appliquer deux méthodes pour ne garder qu'une combinaison des résultats suivant le critère de minimisation de la variance des erreurs (voir paragraphe 2.4). La première méthode de dessaisonalisation est celle des moyennes mobiles. La deuxième est une décomposition par régression linéaire.

La méthode des moyennes mobiles permet d'estimer des coefficients saisonniers selon les 3 étapes suivantes,

- calculer la série des moyennes mobiles centrées,
- calculer l'écart entre les valeurs observées et la moyenne mobile,
- normaliser les écarts, pour aboutir aux coefficients saisonniers.

Cette méthode est appliquée une première fois pour corriger les variations hebdomadaires et une deuxième fois pour corriger les variations journalières. Les résultats obtenus sont les coefficients saisonniers S^1 de l'équation (1).

La deuxième méthode de décomposition est celle proposée par Buys-Ballot (1847). Elle consiste à trouver les coefficients S^2 de l'équation (2) par MCO (Moindre Carrés Ordinaire).

$$ln(U_t^2) - ln(T_t) = \gamma_1 S_t^{2,1} + \gamma_2 S_t^{2,2} + \gamma_3 S_t^{2,3} + \gamma_4 S_t^{2,4} + \gamma_5 S_t^{2,5} + \gamma_6 S_t^{2,6} + \Phi_0 SS_t^{2,0} + ... + \Phi_{52} SS_t^{2,52} + \zeta_t$$

La chronique diminuée de sa tendance, se décompose en une suite de composantes saisonnières correspondantes aux 6 jours de la semaine, aux 53 semaines de l'année et d'un processus

ζ_t. Les p composantes saisonnières sont des variables binaires pour p saisons dans l'année. La variable binaire est égale à 1 lorsque la donnée se rapporte à la saison envisagée et 0 partout ailleurs.

2.2.3 Évènements calendaires

Les évènements suivants sont appliqués sur les deux séries ($(ln(U^i) - ln(T) - ln(S^i)$, $i = \{1, 2\}$), corrigées des variations saisonnières et de la tendance, obtenues par moyennes mobiles et par la méthode de Buys-Ballot. La perte d'un jour d'activité provoque la récupération de cette activité sur les jours voisins. Par exemple, un jeudi férié peut conduire à une augmentation de l'activité le lundi par anticipation ou le vendredi par retard. Souvent, un jour férié a des conséquences prévisibles sur une période de 9 jours (J-4, J, J+4). Les conséquences sont différentes selon le jour férié, le jour de la semaine, l'agence en question.

La récupération de la perte d'un jour de travail sera variable selon que ce jour soit un lundi, un mardi, ou un autre. Si c'est un samedi, il y a peu d'activité à rattraper, alors que le lundi est une journée chargée. Si le férié tombe un vendredi, les GMS anticipent et demandent à être livrées le double jeudi. Les livraisons explosent le jeudi et les expéditions gonflent le mercredi pour des livraisons en A pour B. S'il tombe un lundi les GMS anticipent un peu sur le vendredi précédent et récupèrent surtout le mardi. Si le férié est un jeudi, il y de fortes chances que l'activité du vendredi soit réduite, car les salariés font le pond. Le mercredi précédent sera d'autant plus chargé.

Un jour férié est souvent synonyme de fête ce qui engendre une augmentation de la consommation des ménages et par conséquent un renforcement de l'activité du transporteur. Mais cette hausse est variable selon qu'il s'agisse de la Toussaint ou de Noël. Les agences ne sont pas égales face à la hausse d'activité engendrée par un férié. L'agence de Bretanor par exemple, se trouve être située à côté d'une usine « Côte d'Or ». L'usine approvisionne tous les magasins de France en chocolat de Pâques. Cet approvisionnement commence des mois avant l'évènement et monopolise une grande partie des ressources de l'agence TFE. D'autres n'ont pas ce client et subissent moins de poussée ou elles la subissent à d'autres moments. Canal froid à Nantes par exemple transporte le muguet du 1er avril sur l'Hexagone.

Parfois, le 4^{ime} jour suivant un férié est aussi le 2^{ime} jour précédent un autre férié. C'est ce qui arrive en mai entre la fête du Travail (1er mai) et la Victoire 1945 (8 mai). Dans ce cas, il est difficile de séparer les effets issus des deux fériés.

Pour tenter de séparer ces 4 phénomènes engendrés par la tombée d'un jour férié nous retenons pour chacun d'eux les informations suivantes :

- nom du jour férié,
- jour de la semaine (lundi,..., samedi),
- jour de la semaine des 4 jours précédents et des 4 jours suivants,
- éloignement des 8 jours encadrant le férié.

Il en résulte un tableau de 11 colonnes, une pour identifier le jour de la semaine, une codée entre -4 et 4 pour indiquer la distance du jour impacté au jour férié et les 9 autres pour reconnaître les fériés. Les T lignes représentent le nombre d'enregistrements de l'historique. Cette matrice est convertie en tableau disjonctif complet pour servir de variables binaires afin d'ajuster un modèle de régression sur la série désaisonnalisé : $W = \beta^i F + \xi$, avec $i = \{1, 2\}$, W la chronique désaisonnalisée et F le tableau disjonctif complet. Cette pratique engendre un nombre important de variables (8 voisins $\times$ 9 jours fériés $\times$ 6 jours semaine). Toutes ne sont

pas significatives. Une comparaison entre la variance des estimations et la variance des erreurs (test de Fischer) permet de retenir les variables les plus discriminantes.

2.3 Modélisation des éléments stochastiques

Les procédures précédentes ont permis d'extraire de la chronique la tendance, les composantes saisonnières et les évènements prévisibles. La série temporelle obtenue (V) présente une relation de cause à effet entre l'observation à une date t et les observations précédentes ($t-1$, $t-2$, $t-3$, $t-4$, $t-5$, $t-6$). Pour modéliser et prévoir cette série, nous optons pour le lissage exponentiel simple qui a l'avantage d'être automatisable. $\widehat{V}_{T+1} = (1-\alpha)\sum_{j=0}^{T-1}\alpha^j V_{T-j}$
Il prend en compte les observations passées (de $T-1$ à $T-6$) et les pondère par la constante de lissage α. Cette dernière est estimée de sorte à minimiser la différence carrée entre la chronique et les estimations du modèle.

Les valeurs prédites sont additionnées aux composantes déterministes que nous avions soustraites précédemment. Après application de la fonction exponentielle, nous obtenons l'estimation de la chronique d'origine.

2.4 Combinaison des prévisions

Du fait de deux procédures de décomposition, nous obtenons deux prévisions. L'erreur de prévision obtenue par les deux méthodes ne permet pas de conclure à la supériorité d'une sur l'autre. Le schéma de combinaison choisi, cherche à minimiser la variance de l'erreur de prévision résultant de la performance antérieure des prévisions individuelles (Bourbonnais et Usunier (2007)). Ne dépendant pas de la spécification d'un seul modèle, les prévisions combinées tentent à être plus robustes. La prévision combinée PC, est une moyenne pondérée des deux prévisions individuelles PU^1 et PU^2 ; $PC = \lambda PU^1 + (1-\lambda)PU^2$, λ est le coefficient pondérateur, $0 < \lambda < 1$. Soit EPC, l'erreur de prévision combinée, $EPC = \lambda EPU^1 + (1-\lambda)EPU^2$, la variance de l'erreur de prévision est $V(EPC) = \lambda^2 V(EPU^1) + (1-\lambda)^2 V(EPU^2) + 2(1-\lambda)\lambda COV(EPU^1, EPU^2)$. Nous cherchons λ qui minimise $V(EPC)$, en annulant la dérivée première par rapport à λ. La solution est, si les erreurs de prévisions sont corrélées

$$\lambda = \frac{V(EPU^2) - COV(EPU^1, EPU^2)}{V(EPU^1) + V(EPU^2) - 2COV(EPU^1, EPU^2)}$$

,sinon

$$\lambda = \frac{V(EPU^2)}{V(EPU^1) + V(EPU^2)}$$

2.5 Redressement des prévisions et intervalle de prévision

Les chroniques du poids passé à quai en « expédition », « distribution » et au « total » sont prévues séparément. Or le total doit être égal à la somme des « expéditions » et « distributions ». Pour rendre cohérentes les prévisions, il faut redresser les chiffres. Par leur importance, les chiffres du « total » sont moins variants, donc plus aisés à prévoir. Partant du principe que l'erreur de prévision de la somme est moins élevée que les erreurs cumulées des « expéditions »

et « distributions », nous choisissons de garder la prévision de la série « total des marchandises passées à quai » pour corriger les deux autres. Cette technique est appelée « top-down agrégation ». La procédure d'ajustement est la suivante, soit

- $\widehat{X}_{t+h}$ le poids des expéditions prévues pour la date $t+h$,
- $\widehat{Y}_{t+h}$ le poids des distributions prévues pour la date $t+h$,
- $\widehat{Z}_{t+h}$ le poids total prévu pour la date $t+h$.

Nous cherchons α et β tel que $[\widehat{Z}_{t+h} - (\alpha\widehat{X}_{t+h} + \beta\widehat{Y}_{t+h})]^2$ soit minimum, sous contrainte $1,5 > \alpha > 0,5$ et $1,5 > \beta > 0,5$. Les contraintes indiquent que les prévisions en « expédition » ou « distribution » ne peuvent pas se tromper de plus de 50%. La méthode Quasi-Newton a été choisie pour résoudre le problème d'optimisation.

Pour accompagner les valeurs prévisionnelles, nous calculons les intervalles de prévision bilatéraux au niveau de probabilité de 95%. L'intervalle de prévision peut être utile lorsque qu'une décision est difficile à prendre. Il peut par exemple décider l'attribution d'un camion en plus sur une tournée. D'autre part, l'intervalle de prévision peut donner une idée de la confiance à accorder à la prévision. Plus l'intervalle s'écarte moins la prévision est stable.

3 Applications et résultats

Canal froid est une agence basée à Nantes. 60% de son activité se fait en distribution, contre 40% en expédition. Entre janvier et mars, l'activité est stable autour de 750 tonnes (du lundi au vendredi). Les jours fériés d'avril et mai, provoquent de brusques changements de régime. Le poids des marchandises transportées peut passer de moins de 10 tonnes un jour férié à plus de 1 110 tonnes deux jours avant. Les samedis et jours fériés exclus, l'activité d'avril et mai frôle les 1000 tonnes/jour. Juin est une période d'accalmie. Avec une activité autour de 900 tonnes/jours et quelques jours fériés, juillet et août sont des mois chargés. Septembre, octobre est une période qui retrouve le calme de début d'année avec une moyenne inférieure à 800 tonnes/jour. Enfin, l'activité progresse régulièrement tout le long de décembre pour atteindre un pic de 1 114 tonnes cinq jours avant Noël.

Le tableau 1 compare les résultats obtenus par le modèle avec les observations réelles. L'échantillon d'apprentissage est l'historique des poids précédent la date T. T varie entre le 01/01/07 et le 31/07/07 avec un pas de six jours. Nous prédisons les dates $T+1$ à $T+6$. Nous comparons $\widehat{X}_{T+i}$, $\widehat{Y}_{T+i}$, $\widehat{Z}_{T+i}$ avec X_{T+i}, Y_{T+i}, Z_{T+i} pour $i = \{1, .., 6\}$.

Expédition			Distribution			Total		
RMSE	BAR	DR	RMSE	BAR	DR	RMSE	BAR	DR
25 T*	8%	10%	54 T	9%	12%	61 T	7%	9%

*tonnes

TAB. 1 – *Indicateurs de confiance.*

Les indicateurs d'évaluations sont, la racine de l'erreur de prévision quadratique moyenne (RMSE), le biais absolu relatif (BAR) et la dispersion relative (DR). Les deux derniers indicateurs se réfèrent à la moyenne des valeurs observées. Les résultats du modèle se trompent de

7% par rapport à la moyenne de la série « total des marchandises passées à quai ». Les erreurs sont susceptibles de varier entre plus ou moins 9%. Notons que ces résultats sont meilleurs que ceux obtenus par les modèles existants chez TFE. Cependant, l'objectif d'une erreur de prévision inférieure à 5% n'est pas atteint. De plus, une erreur moyenne de 61 tonnes correspond à plus de 6 camions, ce qui reste important.

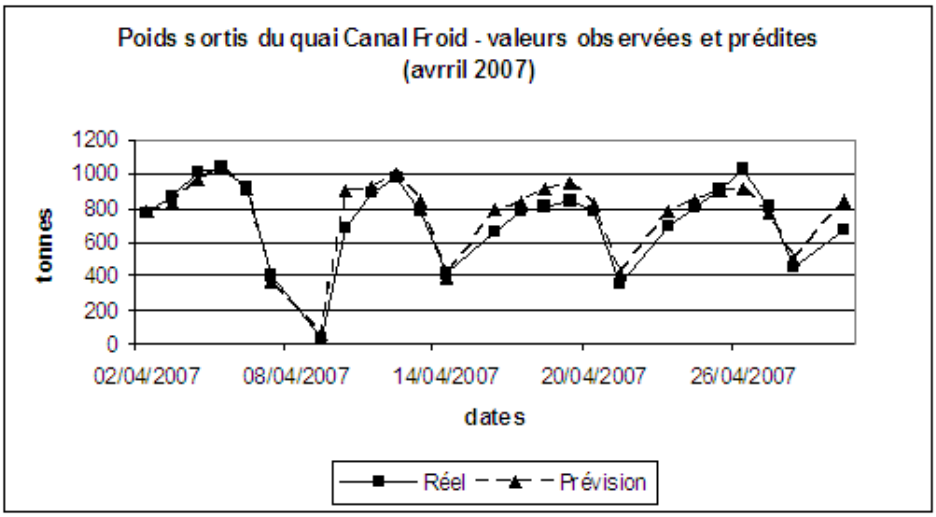


FIG. 1 – *Comparaison observations réelles et prédites*

Ajoutons que la qualité des prévisions varie suivant la bonne stabilité du comportement de l'activité. Une période d'activité stable (janvier à mars), permet d'extrapoler la chronique sans dégager d'erreurs importantes. Par contre, les périodes de turbulence (avril, mai) posent encore des problèmes (voir Fig. 1). Il apparaît que la méthode choisie pour mesurer l'effet des jours fériés sur l'activité n'est pas la plus adéquate. Pour être performante, la méthode nécessite plus d'historique. Rappelons qu'elle estime un coefficient de perte ou gain par MCO pour chacun des 4 jours passés et suivants un férié. Le cycle permettant de retrouver un même jour de la semaine pour un jour férié peut atteindre onze ans. D'autre part, les observations sont éloignées dans le temps et le comportement des ménages change au court du temps.

4 Conclusion et recherches futures

Les calculs de prévision sont centralisés au service d'informatique décisionnelle de STEF-TFE. Les résultats sont publiés sur une interface web spécialement créée pour ce besoin. Ils sont présentés sous forme de tableaux de bords et de graphiques. Les prévisions étant juste à plus de 90% en moyenne, elles sont une sérieuse aide à la décision.

Je suis conscient de la complexité du modèle dû au nombre de paramètres trop important au regard de la taille de l'historique (5 ans). En effet, pour chacune des décompositions il y a 2 paramètres pour la tendance, 318 pour la saisonnalité quotidienne, 53 pour l'hebdomadaire,

20 pour les évènements calendaires et un dernier pour le lisage exponentiel. L'avantage est de pouvoir expliquer au décideur l'effet de chacun des paramètres sur la quantité de marchandises sortie du quai. Une façon de réduire le nombre de paramètres est d'estimer la saisonnalité par une transformée de Fourier et de classifier des évènements calendaires en fonction de leurs effets. Il est également envisagé d'utiliser la corrélation qui peut exister entre les séries temporelles de 57 agences pour améliorer la modélisation. Les corrélations proviennent du fait que les agences TFE forment un réseau et s'expédient de la marchandise entre elles.

Une toute autre approche sera de désaisonnaliser par des filtres de Kalman emboités et introduire dans le modèle, sous forme d'impulsions, les effets des évènements calendaires. C'est une méthode préconisée par Martin (1999) et appliquée sur la prévision de la consommation électrique.

Néanmoins, les économistes misent plus sur ce qu'ils appellent les « prévisions collaboratives ». Les statistiques ne suffisent pas pour obtenir une prévision fiable. Grâce au développement de la gestion partagée, il faut enrichir les résultats par la validation ou les commentaires des différents services de l'entreprise (la logistique, le marketing) et même des clients.

Références

Ayadi, S. (2005). *Le Supply Chain Management : Vers une optimisation globale des flux*. Working paper, Université Catholique de Lyon.

Bourbonnais, R. et J. C. Usunier (2007). *Prévision des ventes, théorie et pratique*. Economica.

Burtschy, B. e. M. C. (1980). A propos de prévision à court terme de la production industrielle. *Revue De Statistique Appliquée tome 28 n^o 2*, 5–24.

Buys-Ballot, C. H. D. (1847). Les changements périodiques de temperature.

Martin, M.-M. (1999). Filtrage de kalman d'une série saisonnière, application à la prévision de la consommation d'électricité. *Revue De Statistique Appliquée tome 47, n^o 4*, 69–86.

Terrolle, C. (2004). *Évolution des rapports entre industriels et grande distribution : du partenariat à la satisfaction clients, vers l´émergence de nouvelles stratégies d´achat*. Mémoire de fin d'études, Université Paris I Panthéon-Sorbonne.

Summary

This article describes a operational research problem. A company that specializes in temperature controlled transportation wants optimize the planning of its human and material resources through short term activity forecasting. The challenge is to find a unique forecasting model adapted, without human intervention, to the specific needs of 57 company's officies. To do it, the company collected data since five years. To analyse them, mathematical algorithms for forecating time series are used. The work is to combine these tools to extract the maximum of determinist information that should be anticipated. The introduction presents the problem and its economic context. This is followed by a description of the process used and arguments to defend choices done. The adopted solutions are inventoried. Finally, the conclusion refers to courses of study.

Applications de gestion de flux de données chez EDF R&D

Sylvain Ferrandiz* ***, Marie-Luce Picard** ***

*GET / Télécom Paris
46, rue Barrault
F-75634 Paris Cedex 13
sylvain.ferrandiz@enst.fr
**EDF R&D
1, avenue du Général de Gaulle
92141 Clamart Cedex
marie-luce.picard@edf.fr
***Laboratoire Commun de Business Intelligence : BILab

Résumé. Les systèmes génériques de gestion de flux de données ont vocation à traiter tout type de flux dès lors que les données sont structurées. A EDF, de nombreuses applications reposent sur l'exploitation de flux issus de domaines variés. Un des objectifs du projet FLUOR, mené au sein d'EDF R&D, est d'étudier la pertinence d'utilisation d'un système de gestion de flux de données générique. L'utilisation d'un système générique pour traiter diverses problématiques permet de réduire les coûts de développement en factorisant les traitements communs (filtrage, agrégation, etc). Nous présentons ici trois applications de démonstration basées sur des thématiques propres à EDF. Ces applications sont développées à l'aide du système commercial StreamBase.

1 EDF, les flux de données et le projet FLUOR

Dans de nombreux domaines, on observe une croissance très importante du volume et du débit des données à traiter : citons par exemple les données transitant sur les réseaux informatiques, les données issues de transactions bancaires et financières, les données issues de réseaux de capteurs.

Cette inflation concerne de nombreux métiers d'EDF : le distributeur devra être à même de gérer, traiter et valoriser les données (comptage, états du réseau...) issues des compteurs communicants (ou : AMM, pour Automatic Metering Management) dont le déploiement massif est prévu d'ici 2016, conformément aux orientations de la Commission de Régulation de l'Energie (`http://www.cre.fr/fr/documents/deliberations`) ; EDF, en tant que commercialisateur, pourra également utiliser ces données de comptage pour imaginer de nouveaux services, pour établir dynamiquement des panels de consommateurs, ou bien encore pour réaliser des prévisions de consommations plus précises ; dans le domaine de la production enfin, l'ensemble des données de fonctionnement, issues pour la plupart de capteurs disséminés au sein des moyens de production, peut être utilisé pour la conduite et la maintenance mais aussi pour la surveillance des matériels.

Face à cette forte augmentation des volumes de données produites par les systèmes de mesure ou les systèmes informatiques, des travaux ont été menés afin de s'affranchir de stocker toutes les données dans une base de données avant de lancer une requête ou de les analyser. Les données sont ainsi traitées lorsqu'elles se présentent, "à la volée". On parle de flux de données, défini comme un ensemble ordonné de données structurées, potentiellement infini.

Le projet FLUOR (Gestion et fouille de flux de données) d'EDF R&D a pour objectif de connaître ces approches de traitement de flux de données, d'évaluer leur intérêt pour les problématiques décisionnelles d'EDF et enfin de développer des prototypes sur ces problématiques. Ces travaux se font en grande partie dans le cadre d'un Laboratoire Commun sur la Business Intelligence (ou Informatique Décisionnelle) contracté en 2007 pour 4 ans avec l'Ecole Nationale Supérieure des Télécommunications (ENST) : le BILab (`http://bilab.enst.fr`).

Dans la section 2, nous présentons succinctement le domaine de la gestion de flux de données. Dans la section 3, nous nous intéressons à un problème de suivi de la consommation électrique avec prédiction et détection des situations de sur-consommation. Dans la section 4, nous présentons une première application de visualisation de la consommation électrique en temps réel. Dans la section 5, nous considérons un problème de suivi de charge et de modélisation de la charge d'un centre d'appels. Enfin, nous concluons à la section 6.

2 La gestion de flux de données

Les données sont récoltées, gérées, analysées, suivant un processus souvent standardisé. L'élaboration des outils nécessaires à la mise en œuvre d'un processus de traitement de données a reposé jusqu'à il y a peu sur un modèle "data pull" : il faut aller chercher les données. Mais de nombreux challenges actuels nécessitent de passer à un modèle "data push" : les données se présentent d'elles-même.

La métaphore du flux se présente alors naturellement pour représenter le mouvement des données. Formellement, un flux de données est une suite de tuples ayant tous la même structure. Cette structure est représentée par un schéma, comprenant le nom des champs du tuple et leur type. La différence entre un flux et une table est le caractère ordonné des tuples. L'ordre est souvent déterminé par un champ d'agencement (typiquement la date, mais pas nécessairement). On entre dans le cadre de la gestion de flux dès lors que

- les données du flux n'ont pas vocation à être stockées,
- les données nécessitent un traitement immédiat,
- les requêtes sont exécutées continuellement (*i.e.* le traitement de flux de données donne naissance à d'autres flux de données).

Tout comme il existe des SGBD pour la gestion de données selon un modèle "data-push", il existe aujourd'hui des systèmes de gestion de flux de données ou SGFD (Babcock et al. (2002), Stonebraker et al. (2005), Carney et al. (2002)). Ceux-ci permettent la définition d'applications renvoyant continuellement des résultats à partir de flux de données les alimentant.

Les SGFD se répartissent en deux catégories : les systèmes spécifiques et les systèmes génériques. Les premiers sont voués au traitement d'un type de données structurées particulier : trafic IP, données de capteurs, etc. Ils exploitent la connaissance de la structure des données. Pour chaque nouveau domaine d'application, il faut utiliser/développer un système particulier. Les SGFD génériques fournissent quant à eux des services de requêtage génériques (filtrage, agrégation) sur des données se présentant sous forme de flux. Ils exploitent uniquement le fait

que les données sont structurées ce qui permet, à travers un unique outil, de traiter des données issues de domaines variés.

Pour démontrer l'intérêt pour EDF d'utiliser un système générique, nous avons choisi d'utiliser StreamBase (`http://www.streambase.com`). Il offre la possibilité de définir graphiquement les applications, en combinant et paramétrant un ensemble de boîtes (les opérateurs graphiques) sous forme d'un graphe orienté acyclique, ce qui facilite sa prise en main et se prête bien à un usage démonstratif.

La réalisation des applications de démonstration passe par le développement d'une application StreamBase et le développement d'une application cliente web. Pour cette dernière, nous utilisons l'outil Flex d'Adobe. StreamBase possède en effet des facilités pour l'interfacer avec une application Flex.

3 Suivi de consommation électrique : prédiction et détection de la sur-consommation

Les données. – On exploite les données de consommation relevée par un compteur électrique. Le schéma des tuples est constitué de deux champs. Le premier est un index de consommation électrique et le second est une estampille temporelle donnant la date du relevé. A partir de données réelles, nous avons construit un flux de données de débit 1 tuple/min.

Les tâches. – D'une part, il s'agit de détecter un dépassement de seuil de consommation. La consommation est la puissance consommée en kW/h. Elle est calculée comme 60 fois la différence entre deux index divisée par la différence entre deux marques temporelles. La détection est réalisée pour chaque marque temporelle multiple de 10 (10h40, 10h50, etc).

D'autre part, on effectue un calcul de la puissance pour chacune des marques intermédiaires (*i.e.* chaque minute) en se basant sur l'index mesuré sur la marque intermédiaire et l'index mesuré sur la marque correspondant à une dizaine précédente. Par exemple, la puissance consommée pour 10h47 est calculée à partir de la différence entre l'index de 10h47 et l'index de 10h40. On considère la puissance obtenue comme une estimation de la puissance consommée à la marque correspondant à une dizaine suivante. Par exemple, la puissance calculée à 10h47 sert d'estimation pour la puissance consommée à 10h50. On envoie une notification (ou une alarme) sur un possible dépassement futur.

L'application. – Le diagramme de l'application StreamBase développée est reporté sur la Fig. 1. Le module GetData, qui regroupe plusieurs traitements, lit les données à partir du fichier, à la vitesse d'un tuple toutes les secondes, et va chercher dans une table l'identifiant du client correspondant au compteur. L'opérateur ReadConnectedClientId lit dans la table ConnectedClient si le client est connecté. Si oui, le module PrepareData évalue un ensemble de champs nécessaires aux traitements subséquents.

L'opérateur Read lit dans la table ReferenceTuple le tuple de référence pour le calcul de la puissance (par exemple, pour le tuple de 10h47, le tuple de référence est celui de 10h40). Le filtre SelectReferenceTuple détecte les tuples de référence, qui sont ensuite stockés dans la table.

Flux de données chez EDF R&D

L'opérateur AveragePower calcule la puissance consommée. Le module GetThreshold lit dans une table la puissance maximum contractualisée par le client, le module DetectOverheadAndDispatch teste s'il y a dépassement et oriente le tuple selon qu'il s'agit d'une détection de dépassement effectif ou d'une prédiction de dépassement. Le module PrepareNotificationDetection (resp. PrepareNotificationPrediction) établit la notification de détection (resp. de prédiction) de dépassement. L'adaptateur Flex assure l'interface avec l'application cliente web en envoyant les notifications vers l'application et en recevant l'identifiant de connection d'un client, aussitôt stocké dans la table ConnectedClient.

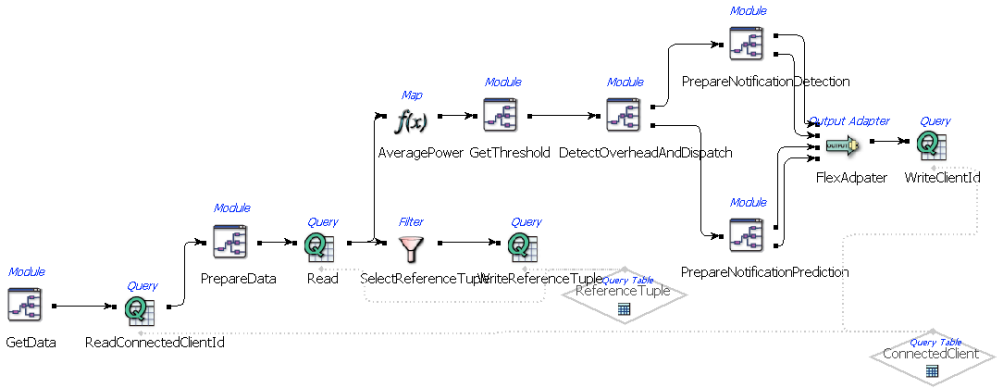


FIG. 1 – *Diagramme de l'application de suivi de consommation.*

Remarque sur le système. – Comme pour tout système générique, les adaptateurs constituent la clé de la généricité de StreamBase. Un *adaptateur d'entrée* lit les données provenant d'une source externe et les place dans un flux d'entrée d'une application. Les données peuvent provenir d'un fichier, d'un socket, d'une base de données, etc. Un *adaptateur de sortie* lit les données provenant d'un flux de sortie d'une application, convertit les données dans un format approprié et les envoie à destination. Ici, nous avons utilisé un adaptateur d'entrée lisant les données dans un fichier et un adaptateur de sortie envoyant les données vers une application cliente web.

Intérêt de la démonstration. – Cette application montre l'intérêt d'exploiter des données fines de comptage. Du point de vue technique, c'est un exemple d'utilisation des adaptateurs (lecture de fichier, alimentation d'un client web) et de fusion d'informations statiques (par exemple issues de Systèmes d'Informations Clients) avec les données du flux (acquisition d'identifiants, de paramètres). Une capture d'écran de l'application web est reportée sur la Fig. 2.

4 Visualisation de consommation électrique en temps-réel

Les données. – On considère ici plusieurs compteurs. Le schéma des données est composé de

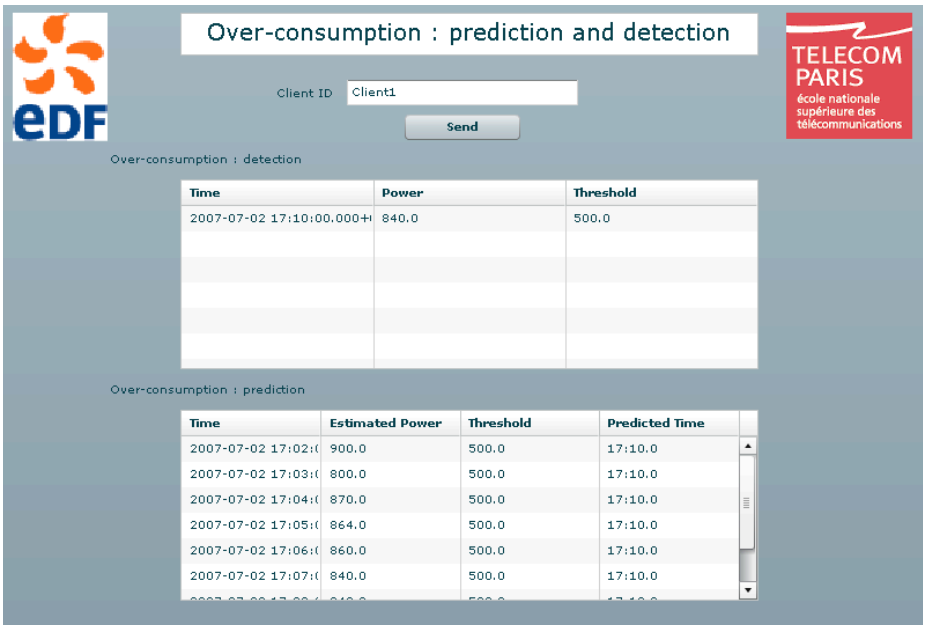


FIG. 2 – *Application web de suivi de consommation. Le client remplit le champ d'identification et clique sur Envoi. Les notifications arrivent ensuite en temps réel.*

trois champs. En plus d'une estampille temporelle et de la puissance électrique consommée, on trouve un champ identifiant le compteur relevé. On dispose d'une mesure de puissance toutes les minutes, sur une période d'un mois et demi et ce pour différents clients. Pour chaque client, nous disposons de données relatives à leurs compteurs.

Les tâches. – L'objectif est de permettre à un client, une fois identifié, de suivre sa consommation électrique en temps réel à travers une application web. Le client doit pouvoir choisir de suivre la consommation mesurée par un compteur particulier ou la consommation globale sur l'ensemble des compteurs. Nous proposons également d'agréger les données sur une fenêtre temporelle, ceci afin de considérer la puissance moyenne consommée sur les dix dernières minutes, par exemple. La taille de la fenêtre est un paramètre utilisateur.

L'application. – Le diagramme de l'application StreamBase développée est reporté sur la Fig. 3. L'application commence par faire entrer les données et adjoint l'identifiant du client au compteur (module GetData, adaptateur TableReader, opérateurs Read et GetClientID, table MeterTable).

La table ConnectedClient stocke les identifiants des clients connectés et est alimentée par l'application web, à travers l'adaptateur Flex. L'opérateur ReadConnection détecte si un client est connecté et le module SumByMeter agrège les données d'un client si celui-ci choisit de visualiser l'ensemble de sa consommation électrique.

Une fois le client identifié, l'opérateur ReadMeterId extrait de la table MeterTable la liste des compteurs du client. Celle-ci, après avoir été mise en forme par l'opérateur PrepareMeterList, est envoyée à l'application web à travers l'adaptateur Flex. Le module AveragePower calcule la puissance moyenne consommée sur une fenêtre temporelle dont la taille est spécifiée par le client.

Toutes les autres opérations ont trait au calcul pour chaque tuple de l'identifiant de la fenêtre dans laquelle il tombe. Il s'agit de recueillir le paramètre utilisateur (opérateur SetWindowSize), de calculer cet identifiant (*i.e.* de l'incrémenter au bon moment) tout en étant robuste face à un éventuel changement de la taille de la fenêtre en cours de travail (opérateurs ReadWindowSize, GenerateWindowId, OnlyUpdateOnChange et UpdateWindowId). La table WindowControl contient toute l'information nécessaire au bon calcul de cet identifiant.

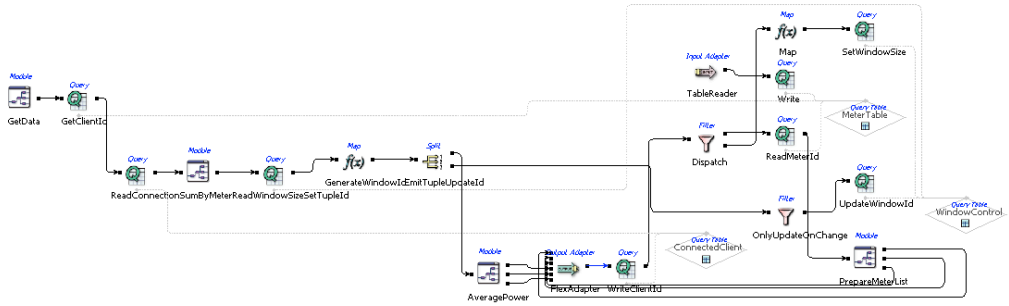


FIG. 3 – *Diagramme de l'application de visualisation de consommation.*

Remarque sur le système. – Les opérations d'agrégation et de jointure sur des flux de données sont potentiellement bloquantes : les flux sont potentiellement infinis. L'idée consiste à travailler sur une portion finie d'un flux : une fenêtre. Dans StreamBase, la gestion des fenêtres est très souple car elle peut être rendue indépendante de l'opérateur utilisé. C'est très utile ici et nous permet de définir une fenêtre dont la taille varie et n'est pas spécifiée initialement.

Intérêt de la démonstration. – Nous avons mis en œuvre une interaction entre l'application web cliente et l'application : identification et acquisition de la liste des compteurs, sélection des compteurs à suivre et spécification de la taille de la fenêtre temporelle. Cela nécessite la mise en place d'un mécanisme de gestion de la dynamicité de la taille de la fenêtre.

Cette application montre la possibilité de proposer des services interactifs aux clients. De plus, c'est un exemple encore plus visuel que le premier de ce qu'apportent les technologies Web 2.0, orientées "data-push", aux applications de gestion de flux de données. Une capture d'écran de l'application web est reportée sur la Fig. 4.

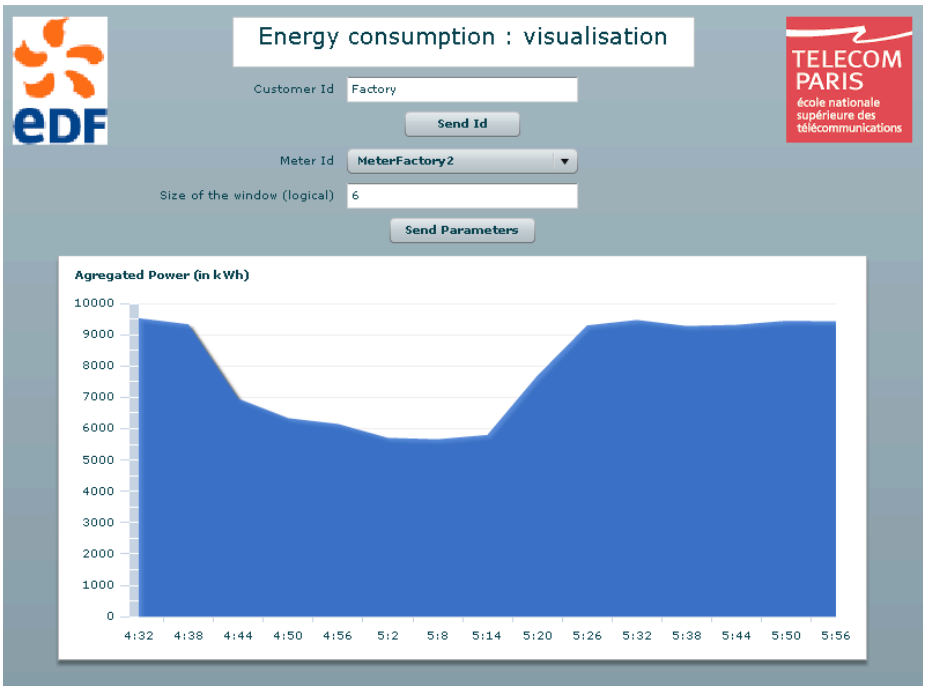


FIG. 4 – *Application web de visualisation de consommation. Le client remplit le champ d'identification et clique sur Envoi. La liste de ses compteurs est alimentée. Il sélectionne le compteur dont il souhaite visualiser les mesures et spécifie la taille logique de la fenêtre pour agrégation. Il clique sur Envoi et la visualisation suit son cours.*

5 Visualisation simultanée de la charge et de la prédiction de la charge d'un centre d'appels

Les données. – On traite ici le nombre d'appels entrants d'un centre d'appels. Le schéma des données est composé d'une estampille temporelle et du nombre d'appels reçus. Nous disposons d'un flux de données sur six mois dont le débit est d'un tuple toutes les cinq minutes, pour chaque journée ouvrée, de 8h à 17h30. De plus, nous disposons d'un modèle linéaire de la courbe de charge permettant une prédiction demi-horaire (8h30, 9h, etc).

Les tâches. – L'objectif est de fournir un outil de supervision pour la gestion du centre d'appels. Une première tâche consiste à visualiser la charge demi-horaire en temps réel ainsi que la prévision du modèle. La seconde vise à détecter les écarts relatifs de prédiction trop grands et à notifier quotidiennement au superviseur le nombre de ces écarts. L'utilisateur spécifie l'écart relatif, en pourcentage, au-delà duquel la prédiction est jugée comme mauvaise. Enfin, nous fournissons une visualisation du nombre d'appels cumulés depuis la demi-heure précédente, toutes les cinq minutes, avec la prédiction du modèle pour la demi-heure suivante. Ainsi, le superviseur voit l'accumulation des appels entre 10h et 10h05, 10h et 10h10 et ainsi de suite

jusqu'à 10h25, et la compare visuellement à la prédiction du modèle pour la tranche 10h-10h30.

L'application. – Le diagramme de l'application StreamBase développée est reporté sur la Fig. 5. L'adaptateur FileDataReader, le module PrepareData et le module GetCoefficientAndComputePrediction font entrer les données dans le système, évaluent les champs nécessaires aux calculs subséquents, acquièrent les coefficients du modèle linéaire, également considérés comme constituant un tuple d'un flux (le modèle pouvant être mis à jour), et calculent la prédiction demi-horaire.

L'opérateur TidyUp2 élimine les champs inutiles et envoie les données vers l'application web cliente à travers l'adaptateur Flex. Ce flux alimente la visualisation de la charge cumulée au pas de cinq minutes.

L'opérateur FilterByMinute filtre les tuples correspondant à une demi-heure (8h30, 9h, etc). Ceux-ci sont envoyés à l'application cliente à travers l'adaptateur Flex après suppression des champs inutiles par l'opérateur TidyUp1. Ce flux alimente la visualisation de la charge demi-horaire réelle et prédite.

L'opérateur ReadWindowId acquière l'identifiant de la fenêtre temporelle à laquelle appartient le tuple en le lisant dans la table WindowId. Cette fenêtre correspond à une journée ouvrée. L'opérateur DetectOverhead évalue si l'erreur relative dépasse un certain seuil et alimente un champ booléen qualifiant (ou non) une notification. Le seuil est stocké dans une variable dynamique alimentée par le flux ErrorRate en provenance de l'application web à travers l'adaptateur Flex. Le module CountDailyAlarm calcule le nombre de notification sur la fenêtre journalière et alimente l'application web à travers l'adaptateur Flex. Le filtre DetectNewDay et l'opérateur UpdateWindowId détectent le changement de journée ouvrée et mettent à jour l'identifiant de la fenêtre temporelle.

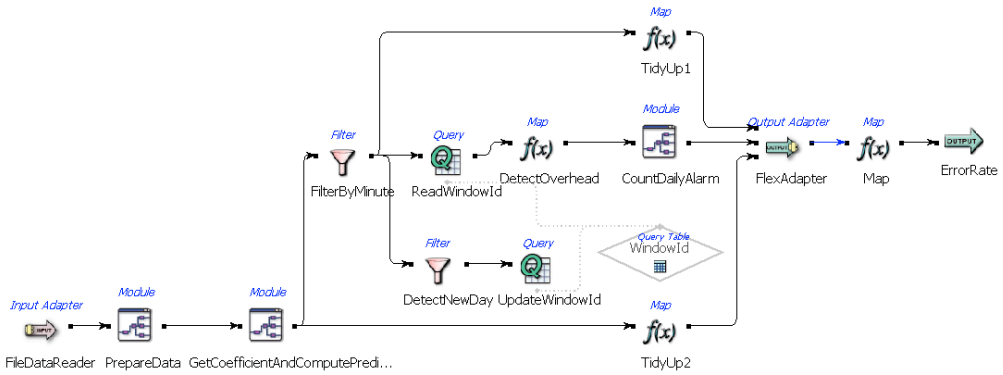


FIG. 5 – *Diagramme de l'application de visualisation de la charge d'un centre d'appels.*

Intérêt de la démonstration. – Cette application est un exemple d'extraction d'informations à différentes échelles de temps : 5 min, demi-heure, journée ouvrée. L'analyse se fait toujours sur des flux de données et en temps réel. Une capture d'écran de l'application web est reportée sur la Fig. 6.

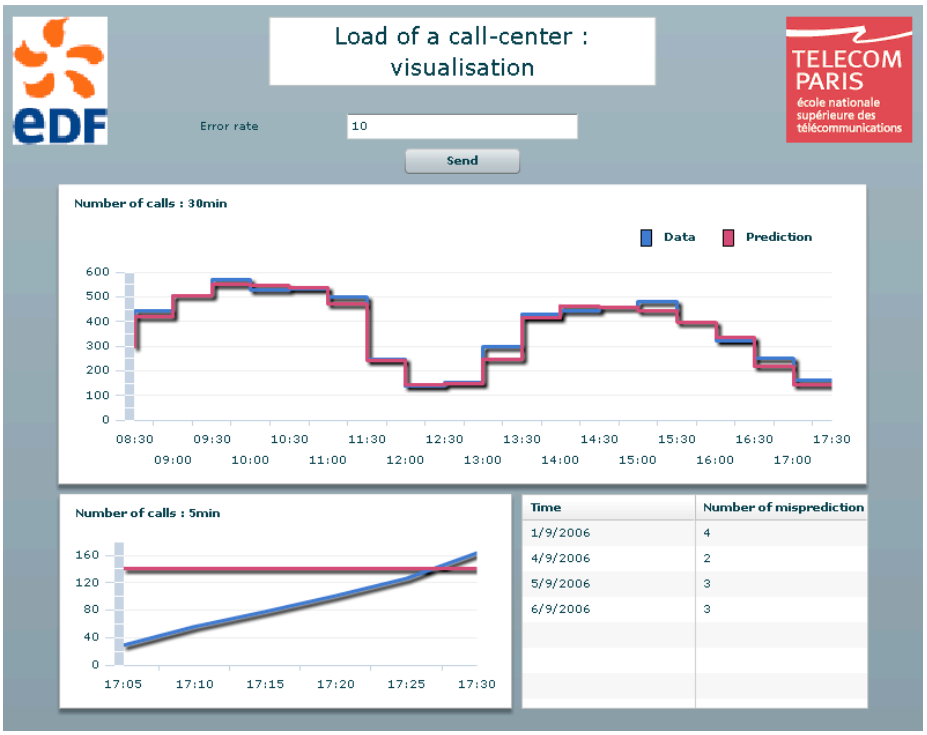


FIG. 6 – *Application web de visualisation de la charge d'un centre d'appels. Le graphique principal montre la courbe du nombre d'appels et la courbe du nombre d'appels estimé à raison d'une donnée toutes les demi-heures. Le second graphique montre le nombre d'appels cumulé depuis le début de la demi-heure, toutes les 5 minutes. Le tableau affiche le nombre de fois sur une journée où l'écart relatif entre la réalité et la prédiction dépasse un certain pourcentage. Le pourcentage est un paramètre spécifié par l'utilisateur dans le champ prévu à cet effet et transmis en cliquant sur Envoi.*

6 Conclusion

A travers ces applications de démonstration, nous avons montré la faisabilité et l'intérêt de traiter en mode "data-push" des données dont le débit et le volume sont potentiellement très importants. Du point de vue applicatif, ces résultats nous permettront de continuer à travailler sur différentes problématiques et avec des données d'origine très variées. C'est là tout l'intérêt de l'utilisation d'un SGFD générique : les traitements mis en œuvre peuvent s'appliquer et/ou s'adapter aisément sur d'autres flux de données, à coûts et durée de développements réduits.

D'un point de vue technique, nous avons investigué différentes possibilités offertes par le SGFD StreamBase, en particulier : les adaptateurs d'entrée nous ont permis de prendre en compte des flux de données arrivant sur un socket, ou des données issues de fichiers ; les adaptateurs de sortie envoient les résultats vers une application cliente Web, ou stockent des

résultats dans un fichier, ou encore émettent des alarmes via l'envoi de courriels. La définition de fenêtres temporelles d'intérêt sur le flux est réalisable aisément et de façon dynamique au cours de l'application. Enfin, des flux de données et des données stockées peuvent être gérées conjointement et à la volée.

Mais de nombreux challenges se présentent encore devant nous. Nous insisterons sur deux d'entre eux.

Pour des raisons internes, nous avons dans un premier temps utilisé en général des données stockées dont on simulait l'arrivée sous forme de flux, dans ce cas particulièrement bien ordonné. La prise en compte de données potentiellement asynchrones nécessitera d'une part une gestion plus "souple" des fenêtres d'intérêt et, d'autre part, l'introduction de fonctions d'approximation dans le calcul d'agrégats ou d'indicateurs à partir de flux distribués. De plus, il s'agit d'évaluer la charge qu'un tel système peut supporter, tant au niveau du débit des flux que de leur nombre, et ce en fonction de la complexité des traitements demandés.

La dernière des applications décrite dans ce papier montre par ailleurs la possibilité de gérer conjointement un flux de données et une modélisation statistique de celui-ci. Dans la maquette réalisée, le modèle linéaire a été mis au point en dehors de l'application, et son évolution n'est pas ou prou gérée. Une application particulièrement intéressante sera la mise au point et la mise à jour d'un modèle de prévision, directement au sein de l'application de traitement de flux. On espère ainsi obtenir des modèles plus précis des données et de leurs évolutions.

Références

Babcock, B., S. Babu, M. Datar, R. Motwani, et J. Widom (2002). Models and issues in data stream systems. In *PODS '02 : proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on principles of database systems*, New York, NY, USA, pp. 1–16. ACM Press.

Carney, D., U. Çetintemel, M. Cherniack, C. Convey, S. Lee, G. Seidman, M. Stonebraker, N. Tatbul, et S. Zdonik (2002). Monitoring Streams - A New Class of Data Management Applications. In *VLDB'02 : proceedings of the twenty-eighth international conference on very large data bases*, Hong Kong, China, pp. 215–226.

Stonebraker, M., U. Cetintemel, et S. Zdonik (2005). The 8 requirements of real-time stream processing. *SIGMOD Rec. 34*(4), pp. 42–47.

Summary

A generic data stream management systems handles any kind of streaming data as soon as they possess a structure described as a schema. In EDF, many applications from various domains, rely on the use of such streams of data. Developping these applications through a generic system is cost saving, by reducing the specific part of the work. The study of the relevance of such a use of a generic system is a part of the FLUOR project in EDF R&D. We present three demonstrative applications issued from EDF. We make use of the commercial StreamBase engine.

Gestion de données et de connaissances pour les bioprocédés

Pascal Neveu *, Virginie Rossard **, E. Aguera ***,
M. Perez *** , C. Picou ***, J.M. Sablayrolles ***

*UMR ASB, INRA, 2 place Viala, 34060 Montpellier
Pascal.Neveu@supagro.inra.fr
**LBE, INRA, avenue des étangs, 11100 Narbonne
Virginie.Rossard@supagro.inra.fr
***UMR SPO, INRA, 2 place Viala, 34060 Montpellier
sablayrolles@supagro.inra.fr

Résumé. Cet article présente une application de gestion d'information dans le domaine des bioprocédés instrumentés (cuve de fermentation, procédés de dépollution, etc). En particulier, nous nous intéressons à la gestion de données issues de mesures sur des fermentations alcooliques et des connaissances associées. Les connaissances, que nous traitons dans ce cadre, sont celles relatives aux événements temporels (opérations et pannes) qui se produisent au cours de l'exploitation de bioprocédés. Les données sont organisées dans une base et nous proposons des organisations rationnelles des connaissances afin d'en permettre une exploitation efficace. Pour cela, nous nous appuyons sur les méthodes et les outils issus du WEB sémantique (XML, Ontologies).

1 Introduction

1.1 Contexte

L'application présentée concerne la gestion des fermentations alcooliques dans un contexte de recherche. Il s'agit d'une application opérationnelle, utilisée par toute une équipe scientifique ainsi que ses partenaires de recherche et industriels. Elle est dédiée à la gestion de données et de connaissances issues de bioprocédés Sablayrolles (2007b). Les cuves de fermentations alcooliques sont instrumentées pour mesurer différentes variables en continu, durant toute la fermentation (au minimum 6 variables toutes les 10 secondes). Les variables les plus importantes sont le débit de CO2, qui caractérise l'activité fermentaire, et la température, qui est un paramètre (que l'on contrôle) essentiel aussi bien vis à vis du déroulement de la fermentation que des caractéristiques organoleptiques du produit Sablayrolles (2007a). D'autres variables figurent dans la base comme la température de consigne, la demande énergétique pour réguler la température, etc. Les moyennes des dernières valeurs des variables, mesurées en ligne, sont stockées dans une base de données avec une fréquence de 20 minutes. Cette fréquence a été déterminée pour appréhender suffisamment finement la dynamique d'une fermentation alcoolique. Les informations essentielles qui caractérisent une fermentation alcoolique sont relatives au moût, à la levure, aux conditions oenologiques. Au cours de ces fermentations, des mesures

hors lignes sont effectuées à partir de prélèvements. Ces mesures peuvent être aussi de type complexe (spectre, comptage, etc).
Les installations différent, évoluent et sont réparties géographiquement. Dans notre cas, nous avons pour l'instant deux sites : Pech Rouge (Aude 11) et Montpellier (Hérault 34) disposant respectivement de 16 cuves de 100 litres et de 30 fermenteurs de 1 litre. Aujourd'hui, nous pouvons donc avoir 46 fermentations qui se déroulent simultanément et dont la duée varie entre 10 et 20 jours. L'Unité Mixte de Recherche des Sciences Pour l'Oenologie (SPO) dispose dans la base de plus de 1500 fermentations archivées. Ce nombre va augmenter considérablement car le nombre de cuves instrumentées ainsi que le nombre de paramètres mesurés en ligne sont sans cesse en augmentation.

1.2 Enjeux et objectifs

Si l'ensemble, composé d'un Système de Gestion de Base de Données relationnel (pour cette application, MySQL) et ses interfaces, est un maillon essentiel, il n'est pas suffisant pour une exploitation efficace avec une finalité de recherche. En effet, il ne fournit pas :

- de cadres formalisés pour les échanges de données,
- de moyens pour la gestion de métadonnées complexes,
- d'environnements pour disposer de connaissances dans le domaine.

Notre objectif est de fournir un Système d'Information efficace pour exploiter et valoriser au mieux les essais. Notre travail a été tout d'abord de concevoir un formalisme XML pour les échanges de données numériques et symboliques. Ensuite nous avons développé un environnement pour la gestion de métadonnées. Cet environnement permet également d'acquérir toutes les informations et la description de pannes et d'opérations sur le bioprocédé. Ces opérations et ces pannes sont des événements qu'il faut prendre en compte. Et enfin, nous avons réalisé un système à base d'ontologies pour la fouille de données et également pour la validation de mesures.

2 Echange de données

Les équipements sur les deux sites concernés sont différents. A titre d'exemple, le débit de CO2 est mesuré soit à partir d'une pesée soit avec un débitmètre. L'instrumentation évolue sans cesse. Par exemple une sonde pH a été ajoutée sur les cuves du centre de Montpellier. Dans le cadre d'un projet européen nous serons amenés à gérer d'autres sites. Il faut donc un moyen générique et souple pour gérer des données temporelles multisources et hétérogènes (voir figure 1). Pour cela XML est une approche efficace pour collecter, gérer et distribuer des données Neveu et al. (2003). Nous avons retenu cette approche pour plusieurs raisons :

- XML est une norme et dispose de nombreux outils (DTD, API, etc),
- XML est indépendant des plateformes et des logiciels,
- les informations sont auto-décrites, voire XML incite à la description,
- XML est une représentation textuelle (par opposition au binaire), donc mieux appréhendée par les personnes Chahuneau (1997),
- XML permet l'organisation de données complexes sous forme de flux et l'intégration des informations temporelles.

Les DTD puis les Schémas nous permettent une validation en ligne de la structure des données Van der Vlist (2002). Cette validation est particulièrement souple et évolutive. Les schémas nous fournissent maintenant la possibilité de valider le contenu en exprimant des contraintes sur les types et les valeurs. Après plusieurs années de mise en oeuvre de XML pour les échanges de données, il est clair que cette approche :

- fournit un cadre et facilite la programmation (SAX, DOM, SimpleXML) comme une procédure très générique d'insertion de données dans une base,
- incite à la description, ce qui est crucial dans les sciences du vivant où beaucoup de paramètres varient,
- augmente la pérennité des données et des applications, en effet XML et ses outils associés permettent d'appréhender :
  1. plusieurs types de procédés et facilitent la prise en compte de nouveaux procédés avec des équipements spécifiques,
  2. la variabilité des types de mesures temporelles effectuées sur un bioprocédé, avec beaucoup de finesse, sous forme de flux.

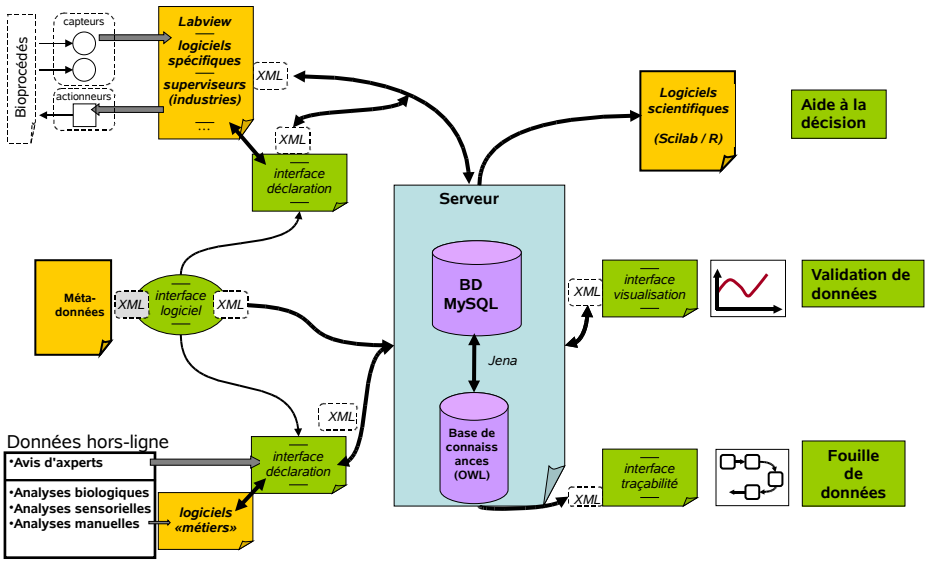


FIG. 1: Architecture du système d'information

Gestion de données et de connaissances pour les bioprocédés

3 Métadonnées

L'utilisation de métadonnées est essentielle pour comprendre et exploiter des données de façon pertinente. Notre premier travail a été la mise en place d'un système autorisant l'annotation des mesures. Pour cela, nous avons défini un schéma XML pour gérer ces mesures et pour permettre de les annoter par différentes applications (acquisitions, calculs de confiance, règles métiers, etc). Ces métadonnées sont inclues dans la base de données. Cependant les métadonnées à structure complexe ne peuvent pas être exploitées de façon efficace avec un SGBD relationnel. En effet un SGBD est mal adapté pour :

- les recherches sur le contenu, par exemple d'un document pdf décrivant un protocole expérimental
- disposer de liens actifs vers différents types de documents (ex : à partir d'un article retrouver les essais utilisées et vice-versa),
- décrire des droits d'accès sans utiliser des fonctions d'administration ou encore intégrer des contraintes juridiques,
- gérer des données symboliques relatives à des événements temporels ou à des expertises d'après leur type ou leur contenu.

Pour mettre en place un système efficace basé sur des métadonnées, celles-ci doivent être fédérées et s'appuyer sur des outils appropriés. Pour cela, nous avons développé des interfaces WEB qui génèrent du XML Rossard (2007). Ces interfaces permettent "d'enrichir" les données à travers plusieurs types de métadonnées que nous avons définies : conditions expérimentales, liens (interne ou externe aux données), interprétations et surtout la description d'événements temporels (opérations, pannes) illustrée à la figure 2 . La gestion de ces événements est essentielle pour la bonne compréhension d'une fermentation. C'est sur ces aspects que nous avons développé plusieurs ontologies. Nous menons également un travail sur la description des ressources (mesures en ligne, mesures hors ligne, commentaires) avec RDF(S) W3C (2004b).

4 Ontologies

Ce travail vise l'intégration des connaissances spécifiques principalement en biologie et en sciences des procédés afin de rendre, une base de données, disponible à une communauté scientifique et technique large et d'en permettre une exploitation efficace. Dans ce but, nous nous appuyons sur les langages et les outils pour le Web Sémantique, même si ils n'apparaissent pas encore pleinement satisfaisants Abel et al. (2005). Nous avons intégré des connaissances dans le système d'information de gestion de bioprocédés. Pour cela, nous avons construit une représentation de celles-ci sous forme d'ontologies que nous formalisons avec OWL W3C (2004a) . Nous disposons de toutes les informations formalisées sur les événements temporels qui se produisent au cours des fermentations. Ces événements sont typés (opération, panne) et pour aller plus loin nous avons constitué plusieurs hiérarchies pour les regrouper. Ces hiérarchies et les relations associées sont directement dépendantes de la finalité recherchée (fouille de données, validation de données, diagnostic, contrôle qualité, etc), leurs utilisations à partir d'un

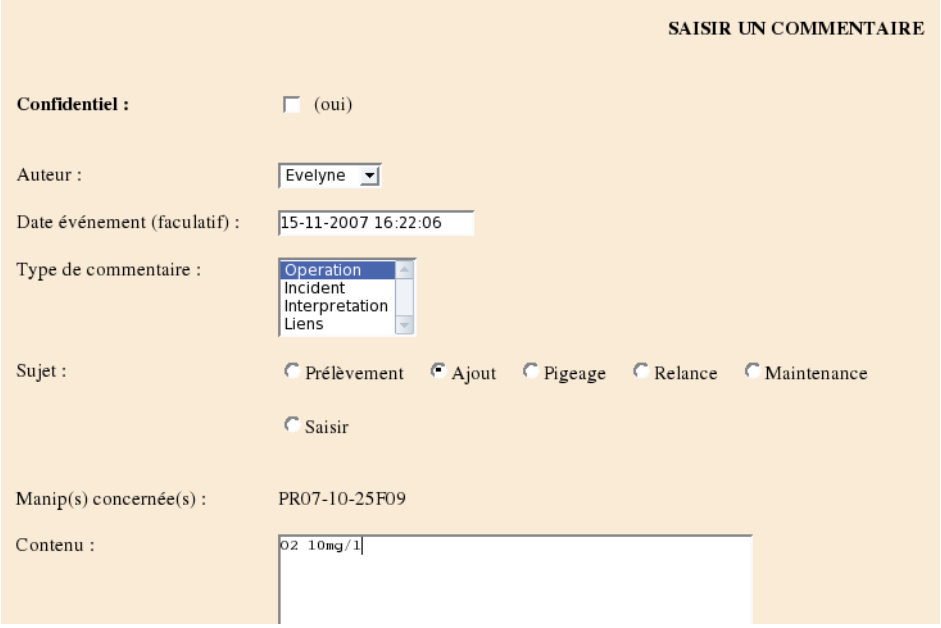


FIG. 2: Déclaration d'un événement temporel

raisonneur (Jena Jena-Team (2000)) nous apportent un nouveau potentiel pour nos développements.

4.1 Gestion de données

Une des grandes préoccupations des utilisateurs (microbiologistes, oenologues, etc) est la confiance qu'ils peuvent accorder aux expérimentations réalisées et surtout si elles sont exploitables pour des travaux d'analyse et de raisonnement menés ultérieurement. Pour répondre à cette attente, nous nous sommes intéressés aux connaissances permettant de déterminer si une fermentation est exploitable ou pas. Plus précisément, si les pannes qui se produisent lors d'une fermentation alcoolique sont rédhibitoires. L'utilisation de ces événements temporels est fondamentale pour permettre aux utilisateurs de fouiller parmi les expériences valides en relation avec la finalité. L'usage des autres événements temporels (opérations) permet d'accroître la puissance des interrogations vers la base de données. La construction de ces moyens d'interrogations complexes est une étape nécessaire pour, par exemple, faire de la classification de courbes. La aussi, les méthodes et outils issus du monde du Web sémantique et des ontologies nous ont permis d'obtenir une organisation adaptée. Cette organisation permet de répondre à des requêtes du type : "quelles sont les fermentations avec une opération de désalcoolisation et qui n'ont pas subi de pannes de régulation". Il existe plusieurs types de pannes de régulation et plusieurs types d'opérations de désalcoolisation, pour répondre à cette requête il faut disposer de ces connaissances (voir figures 3 et 4). Le traitement de requêtes temporelles implicites est réalisé avec le langage R. Cela nous permet de dépasser les limitations liées au SQL.

Gestion de données et de connaissances pour les bioprocédés

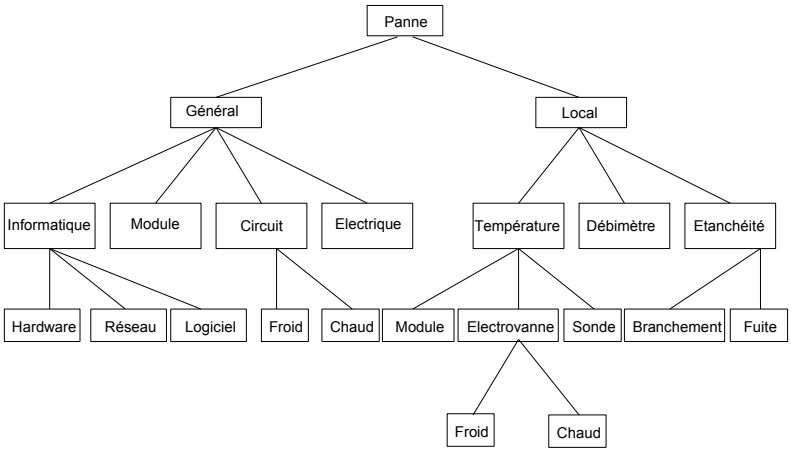


FIG. 3: Arbre des pannes

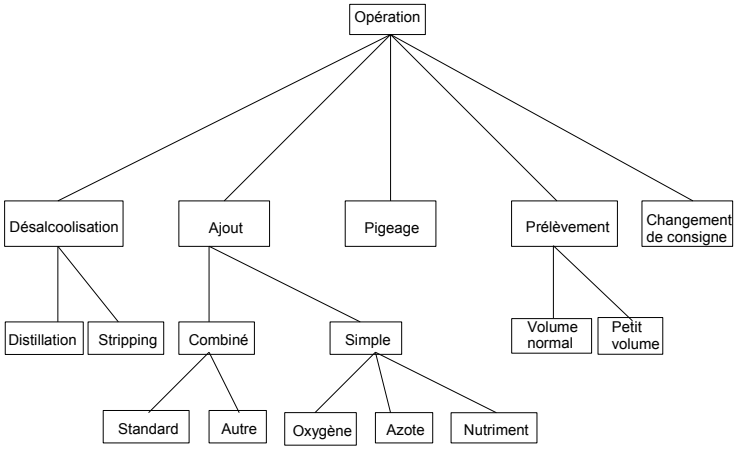


FIG. 4: Arbre des opérations

4.2 Validation de données

Une autre préoccupation est de s'appuyer sur des données qui décrivent correctement l'état du système. Dans notre cas, par exemple les mesures de température ou de débit de gaz peuvent

être fortement perturbées par des opérations ainsi que des pannes. Ces mesures ne reflètent pas la véritable activité fermentaire. Pour répondre à une demande explicite des biologistes concernant la validation de données, nous avons créé une seconde ontologie pour organiser ces événements. En effet, les événements temporels sont les mêmes que précédemment mais c'est la façon dont ces événements impactent les mesures qui nous intéresse pour construire nos regroupements. Par exemple tous les événements qui provoquent une ouverture de cuve courte ou encore tous les événements qui faussent le débit de CO2 (fuite ou problème de débitmètre). Enfin, pour obtenir ces finalités nous avons opté pour une approche hybride s'appuyant sur des fonctions statistiques (R) et l'ontologie qui donne la position temporelle et le choix de l'algorithme.

5 Perspectives et conclusion

5.1 Perspectives

Nos premiers résultats montrent que ces approches méthodologiques et technologiques sont prometteuses dans ce cadre applicatif. Il nous apparaît nécessaire de continuer dans ce sens. Un premier travail est de poursuivre directement sur ces aspects temporels en développant une application basée sur une ontologie spécifique des pannes. Le développement d'une telle application est cruciale par rapport aux coûts humains et financiers engendrés autour de ces expérimentations. Le deuxième travail est la mise au point d'un outil assurant un contrôle de cohérence entre les mesures et les événements temporels. Cet outil devrait avoir un impact important sur la qualité de notre système d'information.

Nous avons la volonté également de mettre en oeuvre cette approche sur des bioprocédés continus comme ceux utilisés en dépollution. La durée d'exploitation de ce type de procédés est de plusieurs années. Ce type de système nous permettra d'éprouver nos réalisations face à une masse d'événements temporels beaucoup plus conséquente et un contexte d'exploitation différent.

5.2 Conclusion

Pour gérer des bioprocédés instrumentés à l'usage d'une communauté scientifique, les systèmes de gestion de base de données ne sont pas satisfaisants. Bien que nous n'ayons pas encore éprouvé la pérennité de cette approche, nous avons montré la faisabilité d'un environnement convivial basé sur les ontologies. Des langages, issus du Web sémantique, comme XML-Schéma, RDF(S), OWL, montrent une efficacité intéressante dans l'exploitation d'expérimentations scientifiques.

Références

Abel, M., R. Dieng-Kuntz, D. Hérin, C. Moulin, P. Pompidor, et T. A. (2005). Langages pour le web sémantique et le e-learning. *Plateforme AFIA / Nice, 30 mai au 3 juin 2005*, 97–122.

Chahuneau, F. (1997). DTDs to XML-Data. *SGML/XML'97 conference procedings*, 337–340.

Jena-Team (2000). Jena - a semantic web framework. Documentation, (http://jena.sourceforge.net/index.html).

Neveu, P., L. Lardon, C. Hacquart, B. Simon, et J.-P. Steyer (2003). PlantML, un langage pour la gestion répartie de bioprocédés. *3ème colloque STIC et Environnement, Rouen, France, 19-20 june 2003*, 197–200.

Rossard, V. (2007). Développement d'un environnement pour la gestion de fermentations alcooliques. Rapport Master, INRA.

Sablayrolles, J. (2007a). Fermented beverages: the example of winemaking. *Advances in fermentation technology*.

Sablayrolles, J. (2007b). Kinetics of yeast fermentation during wine production. *Yeasts in the Production of Wine*.

Van der Vlist, E. (2002). *XML Schema*. O'Reilly.

W3C (2004a). Owl web ontology language. Recommendation, (http://www.w3c.org/TR/owl-features).

W3C (2004b). Rdf vocabulary description language 1.0: Rdf schema. Recommendation, (http://www.w3c.org/TR/rdf-schema).

Summary

This article presents an information management application of instrumented bioprocesses (fermentation bioreactors, waste-treatment processes, etc). Mainly, we are interested in the management of data from these bioprocesses and the associated knowledge. This knowledge is related to temporal events (operation and process failures) that are produced throughout the processes. The data is organized in a database and we propose rational organization of knowledge in order to allow an enficient exploitation. For this reason we utilize methods and tools originally from semantic WEB tools (XML, ontologies).

Influence de l'échantillonnage sur la détection d'objets massifs du trafic Internet

My Huynh Lim, Fabrice Clérot*, Pascal Cheung-Mon-Chan*

* Orange Labs
2, avenue Pierre Marzin BP 50702
22307 Lannion Cedex -France
{fabrice.clerot, pascal.cheungmonchan}@orange-ftgroup.com

Résumé. Le volume des données transitant à travers le réseau Internet nécessite souvent en pratique un échantillonnage préalable de ces données avant leur examen. Cependant, l'influence exacte de cette opération sur certains indicateurs du trafic, comme par exemple la liste des objets massifs du trafic, est mal connue. Aussi, dans cet article, nous étudions quantitativement les conséquence d'un échantillonnage préalable sur la détection des objets massifs du trafic.

1 Introduction

Afin de surveiller le trafic transitant en un point d'un réseau IP, il est souvent utile de connaître la liste des adresses IP source (ou destination) observées en ce point dont la contribution (mesurée par exemple par le nombre de paquets émis par l'adresse considérée) au trafic total pendant une fenêtre d'observation dépasse un seuil donné. En effet, en suivant en permanence ces adresses, qui sont appelées les objets massifs (*Heavy Hitters*) du trafic (Feldmann et al., 2000; Cormode et Muthukrishnan, 2005), il est possible de détecter des variations anormales du trafic et d'alerter si nécessaire un administrateur du réseau.

Cependant, les équipements de réseau actuels ne sont pas suffisamment performants pour capturer tous les paquets IP transitant par les liens à haut débit du réseau Internet et rechercher ensuite les objets massifs du trafic. En pratique, pour estimer la liste des objets massifs du trafic, une démarche courante consiste à échantillonner tout d'abord le trafic puis à rechercher les objets massifs à partir de ce trafic échantillonné. Une telle approche n'est véritablement pertinente que si l'on connait l'ordre de grandeur du taux d'erreurs introduites par l'échantillonnage. Pourtant, à notre connaissance, il n'existe pas dans la littérature scientifique d'études décrivant l'influence de l'échantillonnage sur la détection d'objets massifs. Aussi, l'objectif de cet article est de combler ce manque en étudiant quantitativement les conséquence d'un échantillonnage préalable sur la détection des objets massifs.

2 Préliminaires

2.1 Les objets massifs

Considérons le trafic transitant par un point P d'un réseau IP. On appelle *flot* $\mathcal{F}$ du trafic tout ensemble de paquets transitant par le point P pendant une fenêtre temporelle $\mathscr{W}$ donnée. Par exemple, on peut considérer le flot $\mathcal{F}_a$ des paquets émis par une adresse source a donnée et transitant par le point P pendant la fenêtre $\mathscr{W}$, ou on peut considérer le *flot total* $\mathcal{H}$ qui est formé par la totalité des paquets transitant au point P pendant la fenêtre $\mathscr{W}$. On convient également d'appeler *taille* F d'un flot $\mathcal{F}$ le nombre de paquets du flot $\mathcal{F}$; autrement dit, on a $F = |\mathcal{F}|$. On dit alors qu'un flot $\mathcal{F}$ du trafic est un *objet massif* du trafic pour un seuil relatif $\phi \in [0, 1]$ si et seulement si on a $F \geqslant \phi H$, où F désigne la taille du flot $\mathcal{F}$ et H la taille du flot total $\mathcal{H}$. Par extension, on dit également qu'une adresse source a est un objet massif du trafic pour le seuil relatif ϕ si et seulement si le flot $\mathcal{F}_a$ correspondant à cette adresse source est un objet massif du trafic pour le seuil relatif ϕ. On notera $\mathscr{L}(\phi)$ l'ensemble des adresses source massives du trafic pour le seuil relatif ϕ.

2.2 L'échantillonnage

Dans cet article, nous nous intéresserons à un équipement de surveillance du trafic placé au point P qui échantillonne aléatoirement les paquets du trafic, indépendamment les uns des autres, avec une probabilité λ de tirer un paquet. La probabilité λ est appelée *taux d'échantillonnage*. Un tel équipement transforme donc un flot $\mathcal{F}$ du trafic en un flot échantillonné que nous conviendrons de noter $\overline{\mathcal{F}}$. Nous supposerons également que cet équipement analyse le trafic échantillonné pendant la fenêtre $\mathscr{W}$ et calcule, à partir de ce trafic échantillonné, une liste $\overline{\mathscr{L}}(\phi, \lambda)$ d'adresses source massives pour le seuil ϕ, dans le trafic échantillonné avec un taux λ. Autrement dit, une adresse source a appartient à la liste $\overline{\mathscr{L}}(\phi, \lambda)$ si et seulement si on a $\overline{F_a} \geqslant \phi \overline{H}$, où $\overline{F_a}$ désigne la taille du flot $\overline{\mathcal{F}_a}$ résultant de l'échantillonnage du flot $\mathcal{F}_a$ des paquets émis par l'adresse source a, et $\overline{H}$ désigne la taille du flot $\overline{\mathcal{H}}$ résultant de l'échantillonnage du flot total $\mathcal{H}$.

L'objet de notre étude sera de déterminer dans quelle mesure la liste $\overline{\mathscr{L}}(\phi, \lambda)$, calculée à partir du trafic échantillonné, est proche de la liste réelle $\mathscr{L}(\phi)$ des adresses source massives du trafic non-échantillonné. Pour cela, nous nous intéresserons d'une part à la probabilité de faux positifs (i.e. la probabilité qu'une adresse a n'appartienne pas à $\mathscr{L}$ et appartienne à $\overline{\mathscr{L}}$) et d'autre part la probabilité de faux négatifs (i.e. la probabilité qu'une adresse a appartienne à $\mathscr{L}$ et n'appartienne pas à $\overline{\mathscr{L}}$). Nous étudierons quantitativement l'évolution de ces probabilités d'erreur lorsque l'on fait varier le seuil relatif ϕ et le taux d'échantillonnage λ.

Enfin, nous examinerons dans quelle mesure on peut faire varier le compromis entre la probabilité de faux positifs et la probabilité de faux négatifs lorsque l'on utilise une liste $\overline{\mathscr{L}}(\psi, \lambda)$ avec $\psi = (1 + \alpha)\phi$ pour estimer la liste $\mathscr{L}(\phi)$ (cf § 4.4 pour plus de détails). En particulier, nous tracerons les courbes ROC (*Receiver Operating Characteristics*) obtenues en faisant varier le paramètre α, dit *paramètre de relaxation*.

3 Etude théorique

3.1 Modélisation de la distribution de volume du trafic

Afin d'étudier analytiquement la variation des probabilités d'erreur, nous avons modélisé la distribution de la taille des flots des adresses source[1] du trafic non-échantillonné à l'aide d'une loi de Pareto. Autrement dit, nous avons supposé que la fonction de répartition complémentaire (*complementary cumulative distribution function*) $\gamma(x)$ de la taille F des flots des adresses source est de la forme suivante :

$$\gamma(x) \stackrel{\text{déf}}{=} P(F > x) \tag{1}$$

$$\approx \alpha x^{-\beta} \tag{2}$$

pour tout $x \geqslant 1$ et avec α et β les paramètres de la loi de Pareto. L'utilisation de cette loi pour modéliser la distribution de volume du trafic Internet est classique en métrologie des réseaux (cf par exemple Willinger et al., 1998, ou Guo et al., 2001). En pratique, nous avons constaté que cette loi permet d'approcher de façon satisfaisante la distribution empirique de volume des traces étudiées et nous avons trouvé pour ces données des valeurs de β comprises entre 0,9 et 1,7 (pour plus de détails, cf § 4.2).

3.2 Probabilités d'erreur

Nous allons tout d'abord calculer la probabilité de faux positifs PFP(ϕ, λ), autrement dit la probabilité pour qu'une adresse source a n'appartienne pas à $\mathscr{L}(\phi)$ et appartienne à $\overline{\mathscr{L}}(\phi, \lambda)$. Avec les notations du § 2.2, on a

$$\text{PFP}(\phi, \lambda) = P\left\{a \notin \mathscr{L}(\phi) \wedge a \in \overline{\mathscr{L}}(\phi, \lambda)\right\} \tag{3}$$

$$= P\left(F_a < \phi H, \overline{F_a} \geqslant \phi \overline{H}\right) \tag{4}$$

$$= \sum_{x=1}^{\phi H - 1} P\left(\overline{F_a} \geqslant \phi \overline{H} | F_a = x\right) \times P(F_a = x) \tag{5}$$

Comme les paquets sont échantillonnés indépendamment les uns des autres avec une probabilité de tirage λ, la variable aléatoire $\overline{H}$ suit une loi binomiale d'ordre H et de paramètre λ. Elle a donc pour moyenne λH et pour variance $\lambda(1-\lambda)H$. De même, la loi de la variable aléatoire $\overline{F_a}$ conditionnellement à $F_a = x$ est une loi binomiale d'ordre x et de paramètre λ. Cette loi peut être approchée par une loi gaussienne de moyenne λx et de variance $\lambda(1-\lambda)x$.

[1] On notera que la taille du flot $\mathcal{F}_a$ correspondant à une adresse source a est simplement le nombre de paquets émis par cette adresse.

En négligeant les variations de $\phi\overline{H}$ devant celles de $\overline{F_a}$, on a alors

$$\text{PFP}(\phi,\lambda) \approx \sum_{x=1}^{\phi H-1} P\left(\overline{F_a} \geqslant \phi\lambda H | F_a = x\right) \times P(F_a = x) \tag{6}$$

$$\approx \sum_{x=1}^{\phi H-1} P(F_a = x) \int_{\phi\lambda H}^{+\infty} g_{\lambda x,\lambda(1-\lambda)x}(y)dy \tag{7}$$

$$\approx \int_1^{\phi H} (-\gamma'(x)) \left[\int_{\phi\lambda H}^{+\infty} g_{\lambda x,\lambda(1-\lambda)x}(y)dy\right] dx \tag{8}$$

$$\approx \alpha\beta \int_1^{\phi H} x^{-\beta-1} \left[\int_{\phi\lambda H}^{+\infty} g_{\lambda x,\lambda(1-\lambda)x}(y)dy\right] dx \tag{9}$$

où $g_{\lambda x,\lambda(1-\lambda)x}$ désigne la densité de probabilité de la loi gaussienne de moyenne λx et de variance $\lambda(1-\lambda)x$. En notant Φ la fonction de répartition de la loi normale et en faisant le changement de variable $z = \lambda x$, on peut réécrire l'équation 9 sous la forme suivante

$$\text{PFP}(\phi,\lambda) \approx \alpha\frac{\beta}{\lambda} \int_\lambda^{\phi\lambda H} \left[\frac{z}{\lambda}\right]^{-\beta-1} \Phi\left(\frac{z-\phi\lambda H}{\sqrt{(1-\lambda)z}}\right) dz \tag{10}$$

En négligeant les contributions de l'intégrande pour $z \ll \phi\lambda H$ et en supposant $\phi\lambda H \gg 1$, on obtient finalement l'expression suivante qui met en évidence les dépendances principales de la probabilité de faux positifs

$$\text{PFP}(\phi,\lambda) \approx \alpha\beta(\phi H)^{-\beta-1/2} \left[\frac{\lambda}{1-\lambda}\right]^{-1/2} \int_{-\infty}^{0} \Phi(w)dw \tag{11}$$

On peut également calculer de façon similaire la probabilité de faux négatifs PFN(ϕ,λ), autrement dit la probabilité pour qu'une adresse source a appartienne à $\mathscr{L}(\phi)$ et n'appartienne pas à $\overline{\mathscr{L}}(\phi,\lambda)$. On obtient alors l'expression suivante

$$\text{PFN}(\phi,\lambda) \approx \alpha\frac{\beta}{\lambda} \int_{\phi\lambda H}^{+\infty} \left[\frac{z}{\lambda}\right]^{-\beta-1} \Phi\left(\frac{z-\phi\lambda H}{\sqrt{(1-\lambda)z}}\right) dz \tag{12}$$

En négligeant les contributions de l'intégrande pour $z \gg \phi\lambda H$ et en supposant $\phi\lambda H \gg 1$, on obtient finalement l'expression suivante qui met en évidence les dépendances principales de la probabilité de faux négatifs

$$\text{PFN}(\phi,\lambda) \approx \alpha\beta(\phi H)^{-\beta-1/2} \left[\frac{\lambda}{1-\lambda}\right]^{-1/2} \int_{-\infty}^{0} \Phi(w)dw \tag{13}$$

Cette dernière expression est identique à celle de l'équation 11. Ceci est dû au fait que les approximations qui ont permis d'obtenir les équations 11 et 13 négligent l'asymétrie de la distribution de volume du trafic échantillonné au voisinage de $z = \phi\lambda H$.

4 Résultats Expérimentaux

4.1 Données utilisées pour l'étude expérimentale

Nous avons effectué notre étude expérimentale sur deux traces dont les caractéristiques sont données dans le tableau 4.1. La trace I est disponible publiquement. Il s'agit de la trace Abilene I (NLANR, 2002) qui a été enregistrée par le National Laboratory for Applied Network Research (NLANR) à Indianapolis sur un lien OC-48 (2,5 Gbit/s) du réseau Abilene. Nous avons effectuée notre étude sur les données correspondant au lien vers Cleveland, dans la direction Ouest[2]. La trace II a été enregistrée par France Télécom sur un lien à 1 Gbit/s de son réseau Open Transit IP (OTIP).

Numéro de la trace	I	II
Lien	Abilene (OC-48)	OTIP (1 Gbit/s)
Nombre total de paquets IP	531 M	271 M
Volume de données transporté	343 GB	145 GB
Nombre d'adresses source distinctes	170 K	1.0 M

TAB. 1: Caractéristiques des traces utilisées dans l'étude expérimentale.

4.2 Distribution de volume

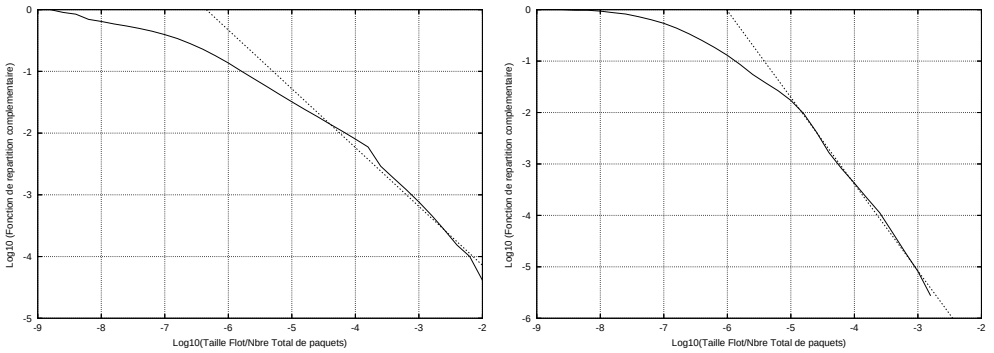
Pour chacune des deux traces étudiées, nous avons tracé la distribution empirique de la taille des flots des adresses source (figure 1). Comme on s'intéresse en pratique à des seuils ϕ qui sont plus grands que 10^{-5}, nous avons recherché le paramètre β de la loi de Pareto qui permet d'approcher au mieux la distribution empirique pour les tailles de flots supérieures ou égales à $10^{-5}H$, où H est le nombre total de paquets de la trace. Pour la trace I, nous avons trouvé $\beta = 0,9$ et pour la trace II, nous avons trouvé $\beta = 1,7$. Nous avons également représenté sur la figure 1 la droite dont la pente correspondant à ces valeurs de β. On peut constater en examinant la figure 1 que la loi de Pareto, avec ces différentes valeurs de β, permet bien d'approcher de façon satisfaisante la distribution empirique de volume des traces étudiées pour des tailles de flots supérieures ou égales à $10^{-5}H$.

4.3 Probabilités d'erreur

Pour chacune des deux traces étudiées, nous avons tracé la probabilité de faux positifs PFP en fonction du seuil relatif ϕ pour différentes valeurs du taux d'échantillonnage λ. L'échantillonnage étant effectué de façon aléatoire, pour chaque valeur du couple (ϕ, λ), nous avons moyenné la probabilité de faux positifs PFP(ϕ, λ) sur 100 épreuves. Les courbes obtenues ont été représentées à la figure 2.

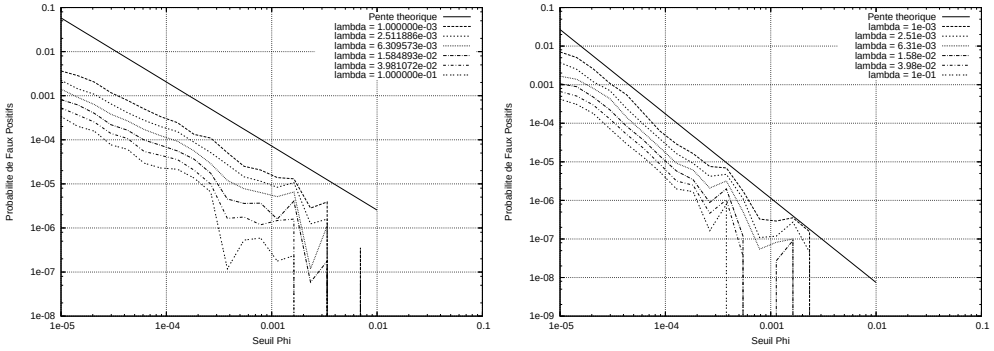
[2]Plus précisément, nous avons utilisé les fichiers de la trace Abilene I de la forme IPLS-CLEV-20020814-xxxx00-0.gz.

Echantillonnage et objets massifs



(a) Trace I (b) Trace II

FIG. 1: Distribution empirique de la taille des flots des adresses source. En pointillé, nous avons tracé une droite dont la pente correspond au paramètre β de la loi de Pareto approchant la distribution empirique.



(a) Trace I (b) Trace II

FIG. 2: Probabilité de Faux Positifs en fonction du seuil ϕ pour différentes valeurs du taux d'échantillonnage λ. Pour chacune des traces, nous avons également tracé une droite de pente $-(\beta + \frac{1}{2})$, le paramètre β ayant été calculé pour chaque trace de la façon indiquée au § 4.2.

Nous avons comparé la façon dont la probabilité PFP(ϕ, λ) dépend empiriquement des paramètres ϕ et λ avec la dépendance prévue par l'expression théorique 11 calculée au § 3.2. Or, pour $\lambda \ll 1$, l'expression 11 peut s'écrire, dans le domaine logarithmique, sous la forme suivante

$$\log(\text{PFP}(\phi, \lambda)) \approx -(\beta + \frac{1}{2}) \log(\phi) - \frac{1}{2} \log(\lambda) + C \tag{14}$$

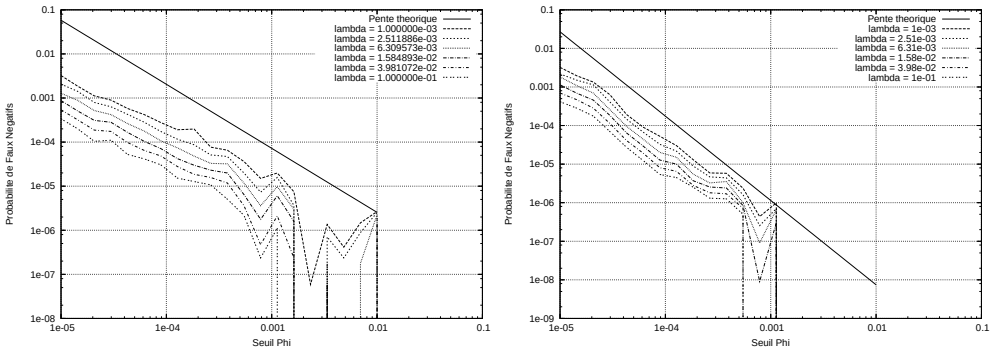
où le terme C est une constante par rapport à ϕ et λ.

D'après l'expression 14, lorsque λ est fixe, la courbe représentant $\log(\text{PFP}(\phi, \lambda))$ en fonction de $\log(\phi)$ doit être une droite de pente $-(\beta + \frac{1}{2})$. Or, en examinant la figure 2, on constate

que, pour chacune des deux traces, lorsque λ est fixe, la courbe représentant $\log(\text{PFP}(\phi, \lambda))$ en fonction de $\log(\phi)$ est effectivement une droite, au moins dans le domaine pour lequel on a $\text{PFP}(\phi, \lambda) \geqslant 10^{-5}$, et la pente de cette droite a effectivement le même ordre de grandeur que la pente théorique prévue pour chaque trace. La dépendance empirique de $\text{PFP}(\phi, \lambda)$ vis à vis de ϕ est donc bien conforme à celle prévue par l'expression 14.

De plus, d'après l'expression 14, pour un seuil ϕ donné, lorsque $\log(\lambda)$ augmente de deux unités, $\log(\text{PFP}(\phi, \lambda))$ doit diminuer d'une unité. Or, en examinant la figure 2, on constate que, pour un seuil ϕ donné, lorsque le taux d'échantillonnage λ augmente de deux ordres de grandeur en passant de $\lambda = 10^{-3}$ à $\lambda = 10^{-1}$, la probabilité $\text{PFP}(\phi, \lambda)$ diminue bien approximativement d'un ordre de grandeur. La dépendance empirique de $\text{PFP}(\phi, \lambda)$ vis à vis de λ est donc bien conforme à celle prévue par l'expression 14.

De même, pour chacune des deux traces étudiées, nous avons représenté sur la figure 3 la probabilité de faux négatifs PFN en fonction du seuil relatif ϕ pour différentes valeurs du taux d'échantillonnage λ en moyennant sur 100 épreuves.



(a) Trace I (b) Trace II

FIG. 3: Probabilité de Faux Négatifs en fonction du seuil ϕ pour différentes valeurs du taux d'échantillonnage λ. Pour chacune des traces, nous avons également tracé une droite de pente $-(\beta + \frac{1}{2})$, le paramètre β ayant été calculé pour chaque trace de la façon indiquée au § 4.2.

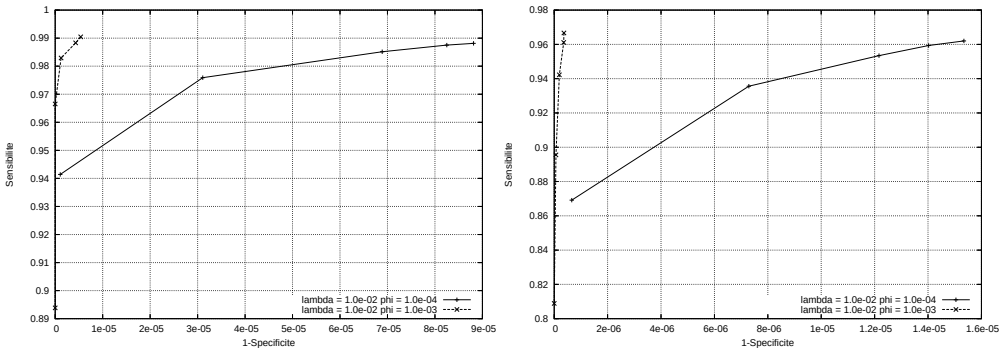
Comme précédemment, nous constatons qu'il y a un bon accord entre la dépendance empirique de la probabilité de faux négatifs $\text{PFN}(\phi, \lambda)$ vis à vis des paramètres ϕ et λ avec celle prévue par l'expression théorique 13.

4.4 Influence d'un paramètre de relaxation

En pratique, il peut être intéressant de faire varier le compromis entre la probabilité de faux positifs et la probabilité de faux négatifs lorsque l'on utilise une liste $\overline{\mathscr{L}}(\psi, \lambda)$ avec $\psi = (1 + \alpha)\phi$ pour estimer la liste $\mathscr{L}(\phi)$.

Afin de comprendre l'influence du paramètre α, dit *paramètre de relaxation*, sur le compromis entre la probabilité de faux positifs et la probabilité de faux négatifs, nous avons tracé à la figure 4 les courbes ROC obtenues empiriquement en faisant varier le paramètre α. Rappelons brièvement qu'une courbe ROC d'un test est le tracé des valeurs de la Sensibilité (*Recall*)

en fonction de 1-Spécificité (*Fallout*) lorsque l'on fait varier un paramètre du test. Dans notre cas, le paramètre que nous avons fait varier est le paramètre de relaxation α, la Sensibilité correspond à la fraction des objets massifs $\mathscr{L}(\phi)$ qui ont été détectés après échantillonnage (autrement dit, qui appartiennent à $\overline{\mathscr{L}}(\psi, \lambda)$) et 1 - Spécificité correspond à la fraction des adresses non-massives (autrement dit les adresses source du trafic n'appartenant pas à $\mathscr{L}(\phi)$) qui sont incorrectement considérées comme des objets massifs après échantillonnage (autrement dit, qui appartiennent à $\overline{\mathscr{L}}(\psi, \lambda)$) .



(a) Trace I (b) Trace II

FIG. 4: Courbes ROC obtenues pour différents seuils ϕ en faisant varier le paramètre de relaxation α. Les différentes valeurs de α étaient (de droite à gauche sur les courbes) 0.001, 0.0032, 0.01, 0.032 et 0.1. Pour l'ensemble des courbes, le taux d'échantillonnage vaut $\lambda = 10^{-2}$.

En examinant la figure 4, on constate qu'en augmentant le paramètre de relaxation α, on diminue 1-Spécificité (autrement dit, le nombre de faux positifs diminue) mais on diminue également la Sensibilité (autrement dit le nombre de faux négatifs augmente). Pour une même diminution de la Sensibilité, on a une diminution de 1-Spécificité d'autant plus importante en valeur absolue que le seuil ϕ est petit. Autrement dit, l'effet du paramètre α est d'autant plus intéressant que le seuil ϕ est petit.

5 Conclusion

Dans cet article, nous avons étudié les conséquences d'un échantillonnage préalable sur la détection des objets massifs du trafic Internet. Nous avons tout d'abord modélisé la distribution de volume du trafic par une loi de Pareto. En utilisant ce modèle, nous avons calculé une expression analytique approchée de la probabilité de faux positifs et de la probabilité de faux négatifs en fonction du seuil ϕ et du taux d'échantillonnage λ. Nous avons ensuite étudié expérimentalement l'influence de l'échantillonnage sur la détection des objets massifs à l'aide de deux traces collectées sur des liens Internet à haut débit. Nous avons vérifié que la distribution de volume de ces deux traces peut être effectivement modélisée par une loi de Pareto. Nous avons tracé expérimentalement les probabilités de faux positifs et de faux négatifs en fonction du seuil ϕ, pour différentes valeurs du taux d'échantillonnage λ. Nous avons obtenu un

bon accord entre nos résultats expérimentaux et nos prédictions théoriques découlant de notre expression analytique des probabilités d'erreurs. Enfin nous avons montré l'intérêt d'utiliser sur le trafic échantillonné un seuil de détection ψ distinct du seuil ϕ correspondant aux objets massifs que l'on veut détecter.

Remerciements

Les auteurs remercient les relecteurs anonymes et Alexis Bondu pour leurs commentaires très pertinents sur une version préliminaire de cet article.

Références

Cormode, G. et S. Muthukrishnan (2005). An Improved Data Stream Summary : The Count-Min Sketch and its Applications. *Journal of Algorithms*.

Feldmann, A., A. Greenberg, C. Lund, N. Reingold, J. Rexford, et F. True (2000). Deriving Traffic Demands for Operational IP Networks : Methodology and Experience. In *SIGCOMM 2000 : Proceedings of the conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, New York, NY, USA, pp. 257–270. ACM.

Guo, L., M. Crovella, et I. Matta (2001). How does TCP generate Pseudo-self-similarity ? *Ninth IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS 2001)*, 215–223.

NLANR (2002). http ://pma.nlanr.net/Traces/long/ipls1.html.

Willinger, W., V. Paxson, et M. S. Taqqu (1998). Self-similarity and Heavy Tails : Structural Modeling of Network Traffic. *A Practical Guide to Heavy Tails : Statistical Techniques and Applications*.

Summary

The massive amount of data carried by the Internet often requires a preliminary sampling of this data before it can be processed. However, the effect of this sampling operation on many traffic indicators, such as the list of the traffic heavy hitters, is not known precisely. For this reason we study here quantitatively the effect of sampling on the detection of the Internet traffic heavy hitters.

I. MuhammadFuadM.M.,Marteau PF

Une Distance d'Édition Etendue Multi Résolution (MREED)

Muhammad Marwan MUHAMMAD FUAD et Pierre-François MARTEAU

VALORIA, Université de Bretagne Sud
BP 573, 56017 Vannes
{marwan.fuad, pierre-francois.marteau}@univ-ubs.fr

Résumé. La représentation symbolique des séries temporelles a attiré beaucoup de chercheurs récemment, car celle-ci est à la base de quelques techniques de réduction de dimensionnalité particulièrement utiles pour accélérer l'indexation et la recherche dans les banques de séries temporelles. Pour améliorer l'efficacité de la recherche de similitude, nous proposons une nouvelle distance adaptée aux séquences symboliques et nous l'évaluons sur des bases de données de série temporelles symbolisées dans le cadre d'une tâche de classification. Nous la comparons à d'autres distances bien connues dans la littérature du traitement symbolique des données. Nous prouvons également, mathématiquement, que notre nouvelle distance est métrique.

1 Introduction

Le problème de la recherche par similarité dans de grandes bases de données a reçu beaucoup d'attention récemment, en raison du grand nombre d'applications que ce type de recherche engendre. La recherche dans ce domaine s'est concentrée sur différents aspects. L'un d'entre eux est la fonction de similarité ou de dissimilarité utilisée pour supporter le mécanisme de la recherche. Bon nombre de distances ou de pseudo-distances ont été suggérées, néanmoins la distance euclidienne est toujours la plus extensivement utilisée, même si celle-ci présente d'inconvénients ; elle est sensible au bruit et au décalage, et ne peut pas traiter les séries temporelles de différentes longueurs ,Wang et al (2005) .

Un autre aspect du problème concerne la représentation des données. De nombreuses techniques ont été également développées pour réduire la dimensionnalité des données en particulier pour le traitement des données temporelles ou séquentielles, comme la transformée de Fourier Discrète (DFT), Agrawal et al.(1993) et Agrawal et al., (1995), La transformée en ondelette discrète (DWT) ,Chan & Fu(1999), la décomposition en valeurs singulières (SVD) ,Korn et al.(1997), l'approximation par segments constants de longueurs adaptés (APCA) ,Keogh et al.(2001), Approximation par segments agrégés (PAA), Keogh et al.(2000) et Yi & Faloutsos (2000), l'approximation par segments linéaires (PLA) ,Morinaka et al.(2001), … etc

Parmi des techniques utilisables pour la compression de données, les représentations symboliques sont potentiellement intéressantes, parce qu'elles permettent de bénéficier de la richesse des algorithmes et des techniques développés pour l'exploitation des bases de données textuelles. Récemment, une méthode particulièrement illustrative a été proposée : SAX ,Lin ,Keogh, Stefano & Bill Yuan-chi (2003). D'un point de vue général, l'approche SAX propose une réduction de la dimension des données en utilisant une technique d'approximation des séries temporelles par segments de taille fixe (PAA). Les approximations sont ensuite discrétisées pour fournir une représentation symbolique en

utilisant un codage de type freeman. Enfin, une mesure de distance est développée pour permettre la recherche par similarité sur les représentations symboliques.

2 La Distance Proposée

2.1 Préambule

Il existe peu de distances qui traitent des données symboliques. Parmi elles, la distance d'édition (ED) ,Wagner & Fischer (1974) est la plus connue. Cette distance est définie par le nombre minimum de suppressions, d'insertions, et de substitutions nécessaires pour transformer une chaîne *S* en une autre chaîne *T*. Cette distance est la mesure de distance principalement utilisée pour comparer deux chaînes symboliques. Malheureusement, la distance d'édition a un inconvénient majeur : elle pénalise toutes les opérations d'édition de la même manière, sans tenir compte du (ou des) symbole(s) concerné(s) par l'opération.. Ceci pose la question de la qualité des similitudes obtenues en appliquant cette distance.

Différentes variantes de cette distance ont été proposées comme la distance d'édition sur des séquences réelles (EDR) ,Chen et al.(2003), et la distance d'édition avec la pénalité réelle (EDRP) ,Chen et al.(2003), la distance d'édition Markov, Wei (2004), ou encore la distance d'édition normalisée Li et Bo (2007).

Dans cet article, nous présentons une nouvelle métrique pour les données symboliquement représentées. Cette nouvelle distance répond en partie aux problèmes mentionnés ci-dessus, et illustrés dans l'exemple ci-dessous, sans ajouter de complexité par rapport au calcul de la distance d'édition.

Considérons l'exemple suivant :

Exemple 1: Etant donnée la chaîne $S_1 = marwan$; en effectuant deux opérations de changement sur S_1 dans les premières et cinquièmes positions nous obtenons la chaîne $S_2 = aarwin$ ..En calculant leur distance d'édition nous obtenons ; $ED(S_1, S_2) = 2$.
Soit NC le nombre de caractères distincts que deux (ou plus) chaînes contiennent, c.-à-d. $NC = \left|\{ch(S_1)\} \cup \{ch(S_2)\}\right|$ Dans notre exemple nous avons : $NC(S_1, S_2) = 6$.

Si dans un deuxième temps nous changeons les mêmes positions dans S_1 avec différents caractères nous obtenons, par exemple, la chaîne $S_3 = barwen$ .En calculant la distance d'édition entre les deux chaînes nous obtenons; $ED(S_1, S_3) = 2$ (qui est identique à $ED(S_1, S_2)$) .Mais nous notons que $NC(S_1, S_3) = 7$. Ceci signifie qu'une opération de substitution a utilisé un caractère qui est plus "familier" aux deux chaînes dans le premier cas que dans le deuxième cas. Autrement dit, S_2 peut être considérée comme étant plus proche de S_1 que de S_3. Cependant, la distance d'édition ne permet pas d'identifier ceci, puisque la distance d'édition reste la même dans les deux cas. Nous verrons plus tard que ce concept de "familiarité" peut être étendu en considérèrent non seulement NC mais également la fréquence des sous-chaînes.

2.2 Définition- La Distance d'Édition Etendue Multi Résolution

Soit A un alphabet fini, et soit $f_i^{(S)}$ la fréquence du symbole i dans S, $ff_{ij}^{(S)}$ est la fréquence des sous-séquences constituées de deux symboles dans S (compris le cas où $i = j$), et soit $f_i^{(T)}$ la fréquence du symbole i dans T, $ff_{ij}^{(T)}$ (compris le cas où $i = j$) est la fréquence des sous-séquences de deux symboles dans T ,et où S , T sont deux chaînes. La distance d'édition étendue multi résolution (MREED) est définie par ;

$$MREED(S,T) = ED(S,T) + \lambda\left[|S|+|T|-2\sum_i \min(f_i^{(S)}, f_i^{(T)})\right] + \delta\left[|S|+|T|-2(\sum_i\sum_j \min(ff_{ij}^{(S)}, ff_{ij}^{(T)})+1)\right]$$

Où $|S|$, $|T|$ sont les longueurs des chaînes S , T et où, $\lambda \geq 0$, $\delta \geq 0$ ($\lambda, \delta \in R$). Nous appelons λ le facteur de fréquence du premier degré, et δ le facteur de fréquence du deuxième degré.

2.3 Théorème :

MREED est une métrique

Preuve:

Avant de prouver ce théorème nous notons facilement que ;

$$\lambda\left[|S|+|T|-2\sum_i \min(f_i^{(S)}, f_i^{(T)})\right] \geq 0 \qquad \forall S,T \tag{1}$$

$$\delta\left[|S|+|T|-2(\sum_i\sum_j \min(ff_{ij}^{(S)}, ff_{ij}^{(T)})+1)\right] \geq 0 \qquad \forall S,T \tag{2}$$

Preuve du théorème : MREED est une distance métrique.

i) $MREED(S,T) = 0 \Leftrightarrow S = T$

i.a) $MREED(S,T) = 0 \Rightarrow S = T$

Preuve: Si $MREED(S,T) = 0$, et tenant compte de (1) et (2), (3),(4) et (5) sont valides ;

$$|S|+|T|-2\sum_i \min(f_i^{(S)}, f_i^{(T)}) = 0 \tag{3}$$

$$\delta\left[|S|+|T|-2(\sum_i\sum_j \min(ff_{ij}^{(S)}, ff_{ij}^{(T)})+1)\right] = 0 \tag{4}$$

$$ED(S,T) = 0 \tag{5}$$

De (5), et puisque ED est une distance, nous obtenons : $S = T$

i.b) $S = T \Rightarrow MREED(S,T) = 0$ (Evident).

de (a) et (b) nous obtenons : $MREED(S,T) = 0 \Leftrightarrow S = T$

ii) $MREED(S,T) = MREED(T,S)$ (trivial)

iii) $MREED(S,T) \le MREED(S,R) + MREED(R,T)$

Preuve : $\forall S,T,R$ Nous avons :

$$ED(S,T) \le ED(S,R) + ED(R,T) \quad (6)$$

(valide puisque ED est une distance métrique). Nous avons aussi :

$$\lambda[|S|+|T|-2\sum_i \min(f_i^{(S)}, f_i^{(T)})] \le \lambda[|S|+|R|-2\sum_i \min(f_i^{(S)}, f_i^{(R)})] + \lambda[|R|+|T|-2\sum_i \min(f_i^{(R)}, f_i^{(T)})] \quad (7)$$

$$\delta[|S|+|T|-2(\sum_i\sum_j \min(ff_{ij}^{(S)}, ff_{ij}^{(T)})+1)] \le \delta[|S|+|R|-2(\sum_i\sum_j \min(ff_{ij}^{(S)}, ff_{ij}^{(R)})+1)] + \delta[|R|+|T|-2(\sum_i\sum_j \min(ff_{ij}^{(R)}, ff_{ij}^{(T)})+1)] \quad (8)$$

(voir l'annexe pour une brève preuve de (7) et (8))

En ajoutant (6), (7) et (8) côte à côte nous obtenons ;

$MREED(S,T) \le MREED(S,R) + MREED(R,T)$.

De i), ii), et iii) nous concluons que le théorème est prouvé.

3 Analyse de Complexité

La complexité en temps de MREED est $O(m \times n)$, où m est la taille de la première séquence et n est la taille de la deuxième séquence, ou $O(n^2)$ si les deux séquences sont des mêmes taille *n*. Cette complexité est grande mais comparable à la complexité de la distance d'édition classique. Par ailleurs, nous devons prendre en compte que MREED est une distance universelle qui peut être appliquée à tous types d'objets ou de données symboliquement représentés, pour lesquels les autres mesures de distance ne sont pas applicables. Afin de réduire la complexité de MREED lorsque l'on cherche à l'appliquer aux séries temporelles, une piste consiste à définir une méthode de représentation symbolique dédiée qui puisse permettre une compression élevée des séries temporelles.

4 Expérimentation

Nous avons examiné notre méthode dans une tâche de classification appliquée sur 12 ensembles de données de type séries temporelles choisis parmi les 20 ensembles de données disponibles sur l'archive d'UCR (réf.).Comme cité précédemment, cette nouvelle métrique est appliquée aux structures de données symboliquement représentées. Nous pensons que le domaine de la bioinformatique ou l'exploitation des bases de données textuelles sont des cadres idéaux pour tester MREED. Néanmoins, puisque notre champ de recherche concerne les séries temporelles, nous avons décidé d'examiner le comportement de MREED dans le contexte de la comparaison de séries temporelles.

Les séries temporelles ne sont pas naturellement représentées de manière symbolique. Mais de plus en plus de chercheurs se sont intéressés à la question du passage du domaine

numérique au domaine symbolique pour la représentation des séries temporelles. Certaines de ces méthodes sont ad hoc, d'autres sont plus sophistiquées. Une des méthodes les plus célèbres dans la littérature est SAX [5]. SAX, en termes simples, se compose de trois étapes; 1-Reduction de la dimensionnalité des séries temporelles en utilisant PAA (après normalisation des séries temporelles). 2-Discrétisation le PAA pour obtenir une représentation discrète des séries temporelles (En utilisant les « breakpoints »).
3-Utilisation d'une distance définie par les auteurs .Pour examiner MREED nous avons procédé de la même manière pour les étapes 1 et 2 ci-dessus pour obtenir une représentation symbolique des séries temporelles, puis dans l'étape 3 nous avons comparé MREED à la distance d'édition (ED), et à SAX. Pour ces trois méthodes, ED, SAX et MREED, PAA (telle que définie dans SAX) a été utilisée pour représenter symboliquement les séries symboliques. Il est important de rappeler que SAX est une méthode qui est conçue directement pour être utilisée sur les séries temporelles, c'est pourquoi elle est très compétitive.

Afin de faire une comparaison juste, nous avons utilisé la même proportion de compression pour les trois méthodes, tel que défini dans SAX (c.-à-d. 1 à 4). Nous avons également utilisé la même taille d'alphabet (3 à 10). ED et SAX sont caractérisés par deux paramètres : la taille d'alphabet et le facteur de compression. MREED est caractérisé par deux paramètres supplémentaires, qui sont le facteur de pondération de fréquence du premier degré : λ , le facteur de pondération de fréquence du deuxième degré. Pour chaque ensemble de données nous testons les paramètres sur l'ensemble d'apprentissage pour obtenir les valeurs optimales de ces paramètres, c'est à dire les valeurs qui réduisent au minimum l'erreur. Puis nous utilisons ces valeurs optimales sur l'ensemble de test (distinct de l'ensemble d'apprentissage) pour établir le taux d'erreur de la méthode utilisée. Quant aux paramètres λ et δ , et pour des raisons de simplification, nous les avons optimisés dans l'intervalle $[0,1]$ seulement, sauf dans les cas où il y a une forte évidence que l'erreur diminue monotonement quand λ ou δ croissent.

A l'issue de l'étape d'optimisation des paramètres sur les ensembles d'apprentissage des 12 bases de données nous avons obtenu les résultats suivants (tableau 1) (il n'y a pas d'apprentissage pour la distance euclidienne). Le meilleur score est accentué.

	La Distance d'Édition (ED)	**(MREED)**	**SAX**
Synthetic Control	0.037	0.033	0.027
Gun-Point	0.02	0.02	0.08
CBF	0.033	0	0.167
Face (all)	0.157	0.157	0.118
OSULeaf	0.2	0.19	0.365
SwedishLeaf	0.34	0.316	0.486
50words	0.253	0.253	0.349
Trace	0.05	0	0.31
Adiac	0.687	0.674	0.918
Yoga	0.193	0.193	0.24
Beef	0.533	0.433	0.467

OliveOil	0.333	0.3	0.833
Erreur Moyenne	0.236	0.214	0.363
Ecart-type	0.210	0.201	0.280

Tableau 1

Nous avons ensuite utilisé les paramètres optimaux pour chaque ensemble de données et pour chaque méthode pour les appliquer sur les ensembles de test. Nous avons obtenu les résultats suivants (tableau 2) ; (*: α est la taille de l'alphabet)

	1-NN Distance Euclidienne	**La Distance d'Édition (ED)**	**(MREED)**	**SAX**
Synthetic Control	0.12	0.037 α* =7	0.053 α =8, λ=0, δ=0.25	0.033 α =10
Gun-Point	0.087	0.073 α =4	0.06 α =4, λ =0.25, δ=0	0.233 α =10
CBF	0.148	0.029 α =10	0.023 α =3, λ =0.25,0.5, δ=0.25	0.104 α =10
Face (all)	0.286	0.324 α =7	0.324 α =7=, λ =0, δ=0	0.319 α =10
OSULeaf	0.483	0.318 α =5	0.302 α =5, λ =0, δ=0.25	0.475 α =9
SwedishLeaf	0.213	0.344 α =7	0.365 α =7, λ =0.25, δ=0	0.490 α =10
50words	0.369	0.266 α =7	0.266 α =7, λ =0, δ=0	0.327 α =9
Trace	0.24	0.11 α =10	0.02 α =6, (λ =0, δ≥0.75), (λ =0,0.25, δ=1)	0.42 α =10
Adiac	0.389	0.701 α =7	0.642 α =9, λ =0.5, δ=0	0.903 α =10
Yoga	0.170	0.155 α =7	0.155 α =7, λ =0, δ=0	0.199 α =10
Beef	0.467	0.467 α =4	0.367 α =4, λ =0.5, δ=0.25	0.533 α =10
OliveOil	0.133	0.467 α =9	0.367 α =9,(λ=0.75,δ≥0.5), (λ =1, δ≥0.75)	0.833 ∀ α
Erreur Moyenne	0.259	0.274	0.245	0.406
Ecart-type	0.138	0.205	0.189	0.265

Table 2

Les résultats obtenus montrent que l'erreur moyenne est la plus petite pour MREED, elle est même plus petite que celle obtenue par la distance euclidienne. Ces résultats montrent également que des trois méthodes examinées (ED, MREED, et SAX) MREED a l'écart type minimum, ce qui signifie que MREED est la plus universelle des trois méthodes examinées. Il faut rappeler que pour la distance euclidienne, il n'y a pas de compression d'information ce qui explique pourquoi celle-ci en moyenne obtient les meilleurs résultats que des distances symboliques compressées

5 Discussion

1-Dans les expériences que nous nous avons effectuées nous avons utilisé des séquences de même taille pour des raisons de comparaison avec la méthode SAX qui ne s'applique qu'aux séquences de la même taille. Mais MREED (ainsi que ED) peuvent être appliquées aux séquences de tailles différentes.

2-Il faut noter nous n'avons pas effectué des expériences avec l'alphabet de taille 2 parce que SAX n'est pas exploitable dans ce cas.

3-Afin de représenter les séries temporelles symboliquement, nous avons dû utiliser une technique de représentation mise en oeuvre pour SAX, ceci à des fins de comparaison. Néanmoins, une technique de représentation dédiée à MREED peut conduire à de meilleurs résultats.

6 Conclusion et perspectives

Dans cet article nous avons présenté une nouvelle métrique appliquée aux séquences. La caractéristique principale de cette distance est qu'elle considère les fréquences des symboles et des sous séquences de deux symboles, caractéristiques que d'autres mesures de distance ne considèrent pas. Nous avons testé cette métrique sur une tâche de classification de séries temporelles, et nous l'avons comparée à deux autres distances (ED et SAX). Nous avons montré que notre distance donne de meilleurs résultats sur les jeux de test considérés.

Une application future possible est d'utiliser MREED pour la détection des motifs dans les séries temporelles, en représentant le motif symboliquement tout en en utilisant la fréquence du motif plutôt que la fréquence des symboles ou des sous- chaînes.

Référence :

Agrawal, R., Faloutsos, C., & Swami, A.: Efficient similarity search in sequence databases". Proceedings of the 4th Conf. on Foundations of Data Organization and Algorithms. (1993)

Agrawal, R., Lin, K. I., Sawhney, H. S. and Shim, K,.: Fast similarity search in the presence of noise, scaling, and translation in time-series databases, in Proceedings of the 21st Int'l Conference on Very Large Databases. Zurich, Switzerland, pp. 490-501(1995).

Chan, K. & Fu, A. W.: Efficient Time Series Matching by Wavelets. In proc. of the 15th IEEE Int'l Conf. on Data Engineering. Sydney, Australia, Mar 23-26. pp 126-133. (1999)

Chen, L., Özsu, M.T. and Oria,V.: Robust and efficient similarity search for moving object trajectories. In CS Tech. Report. CS-2003-30, School of Computer Science, University of Waterloo. (2003)

Jessica Lin, Eamonn J. Keogh, Stefano Lonardi, Bill Yuan-chi Chiu: A symbolic representation of time series, with implications for streaming algorithms. DMKD 2003: 2-11(2003)

Keogh, E,. Chakrabarti, K,. Pazzani, M. & Mehrotra: Dimensionality reduction for fast similarity search in large time series databases. J. of Know. and Inform. Sys. (2000).

Keogh, E,. Chakrabarti, K,. Pazzani, M. & Mehrotra: Locally adaptive dimensionality reduction for similarity search in large time series databases. SIGMOD pp 151-162 (2001)

Korn, F., Jagadish, H & Faloutsos. C.: Efficiently supporting ad hoc queries in large datasets of time sequences. Proceedings of SIGMOD '97, Tucson, AZ, pp 289-300 (1997)

Li Yujian, Liu Bo, "A Normalized Levenshtein Distance Metric," IEEE Transactions on Pattern Analysis and Machine Intelligence ,vol. 29, no. 6, pp. 1091-1095, June, 2007.

Morinaka, Y., Yoshikawa, M. , Amagasa, T., and Uemura, S.: The L lindex: An indexing structure for efficient subsequence matching in time sequence databases. In Proc. 5th PacificAisa Conf. on Knowledge Discovery and Data Mining, pages 51-60 (2001)

Wang Q., Megalooikonomou V., Li G. , A Symbolic Representation of Time Series, Proceedings of the *IEEE Eighth International Symposium on Signal Processing and Its Applications (ISSPA'05)*, Sydney, Australia, Aug. 28-31 , pp. 655-658.(2005)

Wagner, R.,A., Fischer, M. J.: The String-to-String Correction Problem, Journal of the Association for Computing Machinery, Vol. 21, No. I, January 1974, pp. 168--173 (1974)

Wei. J.,Markov Edit Distance. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 26, No. 3, pp. 311—321,(2004).

Yi, B,K., & Faloutsos, C.: Fast time sequence indexing for arbitrary Lp norms. Proceedings of the 26st International Conference on Very Large Databases, Cairo, Egypt (2000)

UCR Time Series Classification/Clustering Page
http://www.cs.ucr.edu/~eamonn/time_series_data/

Annexe

(En raison de la limitation de l'espace nous présenterons sommairement notre preuve)

LEMME :

Soit A un alphabet fini, et soit $f_i^{(S)}$ la fréquence d'un caractère i dans S , où S est une séquence représentée par A.

Soit: $D(S,T) = |S| + |T| - 2\sum_i \min(f_i^{(S)}, f_i^{(T)})$

$\forall S_1, S_2, S_3$ nous avons:

$$D(S_1, S_2) \leq D(S_1, S_3) + D(S_3, S_2) \qquad (1)$$

Quel que soit n, où n est le nombre de symboles utilisés pour représenter les séquences.

Preuve

Nous prouverons le lemme ci-dessus par induction.
i) Cas de base : $n = 1$; c'est un cas insignifiant. $\forall S_1, S_2, S_3$ représentées par le même symbole a
Soit S_1^a, S_2^a, S_3^a les fréquences du symbole a dans S_1, S_2, S_3, respectivement. Dans ce cas nous avons six configurations :

$1)\ S_1^a \leq S_2^a \leq S_3^a$, $2)\ S_1^a \leq S_3^a \leq S_2^a$, $3)\ S_2^a \leq S_1^a \leq S_3^a$
$4)\ S_2^a \leq S_3^a \leq S_1^a$, $5)\ S_3^a \leq S_1^a \leq S_2^a$, $6)\ S_3^a \leq S_2^a \leq S_1^a$

Nous prouverons que relation (1) se tient dans ces six configurations.
$1)\ S_1^a \leq S_2^a \leq S_3^a$
Dans ce cas nous avons : $\min(S_1^a, S_2^a) = S_1^a$, $\min(S_1^a, S_3^a) = S_1^a$, $\min(S_2^a, S_3^a) = S_2^a$

$$D(S_1, S_2) \overset{?}{\leq} D(S_1, S_3) + D(S_3, S_2)$$

En substituant les valeurs ci-dessus dans cette dernière relation nous obtenons :

$$S_1^a + S_2^a - 2S_1^a \overset{?}{\leq} S_1^a + S_3^a - 2S_1^a + S_3^a + S_2^a - 2S_2^a \Rightarrow 0 \overset{?}{\leq} 2S_3^a - 2S_2^a$$

Ce qui est valide compte tenu des conditions qui caractérisent cette configuration.
Les preuves des cas 2), 3), 4), 5) et 6) sont similaires à la preuve du cas 1).
De 1)-6) nous concluons que le lemme est valide pour $n = 1$

ii) Cas d'induction : nous supposons que le lemme se tient pour $n-1$, où $n \geq 2$ et nous le prouvons pour n. Puisque le lemme se tient pour $n-1$ nous avons :

$$D(S_1, S_2) \leq D(S_1, S_3) + D(S_3, S_2) \qquad (2)$$

où:

$$D(S_1, S_2) = |S_1| + |S_2| - 2\sum_{i=1}^{n-1} \min(f_i^{(S_1)}, f_i^{(S_2)})$$

$$D(S_1, S_3) = |S_1| + |S_3| - 2\sum_{i=1}^{n-1} \min(f_i^{(S_1)}, f_i^{(S_3)})$$

$$D(S_3, S_2) = |S_3| + |S_2| - 2\sum_{i=1}^{n-1} \min(f_i^{(S_3)}, f_i^{(S_2)})$$

Soient $f_n^{(S_1)}, f_n^{(S_2)}, f_n^{(S_3)}$ les fréquences du symbole nouvellement ajouté pour représenter les séquences S_1, S_2, S_3 respectivement. Nous avons six configurations pour le symbole nouvellement ajouté

7) $f_n^{(S_1)} \le f_n^{(S_2)} \le f_n^{(S_3)}$, 8) $f_n^{(S_1)} \le f_n^{(S_3)} \le f_n^{(S_2)}$, 9) $f_n^{(S_2)} \le f_n^{(S_1)} \le f_n^{(S_3)}$

10) $f_n^{(S_2)} \le f_n^{(S_3)} \le f_n^{(S_1)}$, 11) $f_n^{(S_3)} \le f_n^{(S_1)} \le f_n^{(S_2)}$, 12) $f_n^{(S_3)} \le f_n^{(S_2)} \le f_n^{(S_1)}$

Nous prouvons que relation (1) se tient dans ces six configurations.

7) $f_n^{(S_1)} \le f_n^{(S_2)} \le f_n^{(S_3)}$. Dans ce cas nous avons:

$$\min(f_n^{(S_1)}, f_n^{(S_2)}) = f_n^{(S_1)}, \min(f_n^{(S_1)}, f_n^{(S_3)}) = f_n^{(S_1)}, \min(f_n^{(S_2)}, f_n^{(S_3)}) = f_n^{(S_2)}$$

$$D(S_1, S_2) \overset{?}{\le} D(S_1, S_3) + D(S_3, S_2) \Rightarrow$$

$$|S_1| + |S_2| - 2\sum_{i=1}^{n-1} \min(f_i^{(S_1)}, f_i^{(S_2)}) + f_n^{(S_1)} + f_n^{(S_2)} - 2f_n^{(S_1)} \overset{?}{\le}$$

$$|S_1| + |S_3| - 2\sum_{i=1}^{n-1} \min(f_i^{(S_1)}, f_i^{(S_3)}) + f_n^{(S_1)} + f_n^{(S_3)} - 2f_n^{(S_1)} +$$

$$|S_3| + |S_2| - 2\sum_{i=1}^{n-1} \min(f_i^{(S_3)}, f_i^{(S_2)}) + f_n^{(S_3)} + f_n^{(S_2)} - 2f_n^{(S_2)}$$

$$\Rightarrow$$

$$|S_1| + |S_2| - 2\sum_{i=1}^{n-1} \min(f_i^{(S_1)}, f_i^{(S_2)}) \overset{?}{\le}$$

$$|S_1| + |S_3| - 2\sum_{i=1}^{n-1} \min(f_i^{(S_1)}, f_i^{(S_3)}) +$$

$$|S_3| + |S_2| - 2\sum_{i=1}^{n-1} \min(f_i^{(S_3)}, f_i^{(S_2)}) + 2f_n^{(S_3)} - 2f_n^{(S_2)}$$

Tenant compte de (2), nous obtenons: $0 \overset{?}{\le} 2f_n^{(S_3)} - 2f_n^{(S_2)}$,qui est valide selon (7).

Les preuves des cas 8), 9), 10), 11) et 12) sont similaires à la preuve du cas 7).

De 7)-12) nous concluons que le lemme est valide pour n.

De i) et ii), nous concluons que le lemme est valide

L'épreuve de (8) dans le théorème est identique à cette de (7)

Summary

Symbolic representation of time series has attracted many researchers recently, since it is at the basis of some dimensionality reduction techniques particularly useful to speed up time series indexing and retrieval. To improve the effectiveness of similarity search, we propose an extension to the edit distance metric that is applied to symbolic sequential data objects and we test it on time series data bases in some classification task experiments. We compare it to other distances that are well known in the literature for symbolic data objects. We also prove, mathematically, that our extended distance is a metric.