Prédiction de Liens Temporels en Intégrant les Informations de Contenu et de Structure

Sheng Gao, Ludovic Denoyer, Patrick Gallinari

Université Pierre et Marie Curie - LIP6
4 place Jussieu, 75005 Paris, France
{sheng.gao, ludovic.denoyer, patrick.gallinari }@lip6.fr

RÉSUMÉ. *Dans cet article nous traitons le problème de la prédiction de liens temporels, qui consiste à prédire l'apparition des nouveaux liens dans des graphes de données dynamiques. Cette tâche apparaît dans les applications telles que la recommandation ou l'analyse des réseaux sociaux. La plupart des approches de prédiction de liens temporels se basent uniquement sur la structure topologique globale du réseau. Nous proposons ici un modèle qui exploite simultanément différentes sources d'information dans le réseau afin de prédire les probabilités d'occurrence de liens futurs. Le modèle intègre trois types d'informations : la structure topologique du réseau, le contenu des noeuds dans le réseau et l'information de proximité locale des noeuds. Il basé sur une formalisation à base de factorisation en matrices non négatives et de régularisation dans les graphes. Nous proposons une méthode d'optimisation alternée efficace pour l'apprentissage. Nous expérimentons enfin cette méthode sur plusieurs ensembles de données du monde réel et montrons que notre modèle surpasse les méthodes de l'état de l'art.*

ABSTRACT. *In this paper we address the problem of temporal link prediction, i.e., predicting the apparition of new links, in time-evolving networks. This problem appears in applications such as recommender systems, social network analysis or citation analysis. Link prediction in time-evolving networks is usually based on the topological structure of the network only. We propose here a model which exploits multiple information sources in the network. The model integrates three types of information: the global network structure, the content of nodes in the network if any, and the local or proximity information of a given vertex. The proposed model is based on a matrix factorization formulation of the problem with graph regularization. We derive an efficient optimization method to learn the latent factors of this model. Extensive experiments on several real world datasets suggest that our unified framework outperforms state-of-the-art methods for temporal link prediction tasks.*

MOTS-CLÉS : *Prédiction de liens temporels, Factorisation matricielle.*

KEYWORDS: *Temporal Link Prediction, Matrix Factorization.*

1. Introduction

La tâche de prédiction de liens [GET 05] est une tâche générique présente dans de nombreuses applications comme le Marketing Viral, la recommandation ou l'analyse de réseaux sociaux. Elle a surtout été considérée du point de vue du réseau statique, et consiste alors à inférer des liens manquants à partir de la connaissance d'un réseau partiel [GET 05]. Cependant, les réseaux réels sont généralement dynamiques : à chaque instant, de nouvelles relations entre les entités déjà présentes dans le réseau ou des entités nouvelles peuvent apparaître. Par exemple, on peut être intéressé à conjecturer si deux individus seront dans le futur reliés par un certain type de relation, et ce même si aucune relation n'a été observée entre eux dans le passé. Dans cet article nous étudions le problème de la prédiction de liens temporels : à partir d'un réseau observé pendant T instants, peut-on prédire la structure des liens futurs à $T + 1$? Cette tâche est illustrée sur la Figure 1.

Plusieurs approches ont été proposées dans ce cadre. La plupart des approches, comme le travail de pionnier de [LIB 03], ou plus tard les papiers [HUA 09] et [ACA 09] ne considèrent que l'information sur la structure topologique du réseau pour effectuer la prédiction de liens. Cette information peut se révéler insuffisante, particulièrement dans le cas où il y a peu de données observées. Par exemple dans le cas de réseaux sociaux qui sont en général très peu denses il est nécessaire d'utiliser des informations additionnelles pour capturer la dynamique de création des liens.

Certains auteurs ont posé le problème comme une classification binaire pour prédire la création future d'un lien [WAN 07] et utilisent conjointement la structure topologique du réseau et les attributs (contenu) des noeuds en entrée des classifieurs. Les données sont souvent mal adaptées à ces approches classification à cause de la faible densité des réseaux qui limitent les données disponibles pour apprendre et surtout à cause du déséquilibre des classes, le nombre de paires de noeuds liés est en général de plusieurs ordres de grandeur inférieur à celui des paires de noeuds non liés.

Afin d'apporter une solution à ces problèmes, nous proposons un modèle basé sur une formalisation de factorisation matricielle régularisée qui exploite simultanément plusieurs sources d'information dans le réseau pour prédire les probabilités d'occurrence des liens futurs. Le modèle intègre trois types d'information : la structure topologique globale du réseau, le contenu des noeuds dans le réseau le cas échéant, et l'information de proximité topologique locale des noeuds. L'utilisation conjointe de plusieurs sources d'information et l'analyse matricielle régularisée apporte une solution aux problèmes de données clairsemées et aux problèmes de classes déséquilibrées.

Nos contributions sont les suivantes : (1) nous proposons un cadre pour l'intégration de multiples sources d'information pour la prédiction de liens temporels. Pour cela, nous exploitons une méthode de factorisation matricielle et proposons un algorithme d'optimisation alterné efficace pour l'apprentissage des facteurs latents du modèle. (2) Nous testons cette méthode sur des ensembles de données du monde réel pour des problèmes de co-citation et la comparons avec plusieurs méthodes de l'état

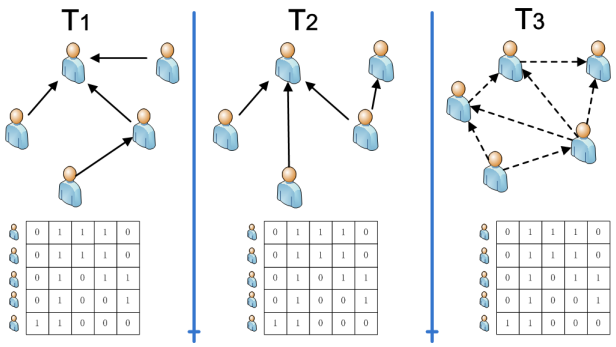


Figure 1. *La tâche de prédiction de liens temporels dans les réseaux dynamiques. Etant donné deux observations du réseau aux temps T_1 et T_2, nous voulons prédire la structure du réseau au temps T_3.*

de l'art. Ces expériences montrent que sur ces données, la méthode proposée dépasse les méthodes de référence.

Dans la Section 2 nous présentons brièvement la formulation du problème et les notions de base concernant la factorisation matricielle et la régularisation dans les graphes. Le modèle est décrit en Section 3. Les expérimentations et les comparaisons avec l'état de l'art sont exposées en Section 4.

2. Description du Problème et Prérequis

2.1. *Prédiction de Liens Temporels*

La prédiction de liens temporels est définie de la manière suivante[1] : on considère un graphe $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, où $\mathcal{V} = \{x_i\}_{i=1}^{N}$ est l'ensemble des noeuds et $\mathcal{E}$ est l'ensemble des liens. Chaque noeud x_i a un contenu décrit par un vecteur de caractéristiques $\mathbf{C}_i = \{C_{ij}\}_{j=1}^{M}$. La matrice de contenu associée au graphe $\mathcal{G}$ sera notée $\mathbf{C} \in \mathbb{R}^{N \times M}$. Nous considérons que les noeuds ainsi que leur contenu ne changent pas au cours du temps et que seule la structure topologique du réseau - les liens - évolue dynamiquement. Une série temporelle de graphes $\{\mathcal{G}_1, ..., \mathcal{G}_T\}$ sera ainsi représentée par une série de matrices d'adjacence $\{\mathbf{A}_t, t = 1, ..., T\}$, où $\mathbf{A}_t \in \mathbb{R}^{N \times N}$. Pour une paire de noeuds x_i et x_j, $\mathbf{A}_t(i, j) = 1$ si il existe un lien entre x_i et x_j au temps t, et $\mathbf{A}_t(i, j) = 0$ sinon. $\mathbf{A}_t$ peut être une matrice symétrique ou non, selon que le graphe est orienté ou non-orienté.

Etant donnée une série de T matrices d'adjacence $\{\mathbf{A}_1, ..., \mathbf{A}_T\}$ et une matrice de contenu $\mathbf{C}$, le problème de la prédiction temporelle de liens correspond à l'estimation de la probabilité d'occurrence de l'ensemble des liens possibles à l'instant $T+1$. Nous

1. Nous utilisons la définition de [HUA 09]

allons noter $\mathbf{Sc}$, la matrice $N \times N$ des scores prédits où $\mathbf{Sc}(i,j)$ correspond au score prédit pour le lien (i,j). Plus ce score est élevé, plus la probabilité que le lien existe à l'instant $T+1$ est importante : $\mathbf{Sc}_{T+1} = f(\mathbf{A}_1, ..., \mathbf{A}_T, \mathbf{C})$, où f correspond au modèle de prédiction.

2.2. *Modèles à facteurs latents*

Le modèle proposé dans ce papier permet l'intégration de différentes sources d'information. Il est basé sur le travail de [ZHU 07] qui propose un formalisme de factorisation matricielle permettant la combinaison de l'information de contenu et de l'information de liens, ainsi que sur les travaux sur la régularisation dans les graphes pour le filtrage collaboratif [GU 10] permettant la prise en compte de l'information locale de proximité entre noeuds d'un graphe. Notre méthode combine ces idées en un modèle unique. Nous décrivons brièvement ci dessous les idées de base puis introduisons notre modèle en section 3.

2.2.1. *Modèle de Factorisation Latent pour la Combinaison des Liens et du Contenu*

Zhu et al. [ZHU 07] a proposé un modèle supervisé de factorisation matricielle permettant la combinaison de l'information de liens et de contenu pour la classification. Les deux types d'information sont projetés dans un espace latent commun de faible dimension. Soit $\mathbf{A}$ et $\mathbf{C}$ respectivement la matrice d'adjacence et la matrice de contenu du graphe. Le problème de factorisation est formalisé de la manière suivante :

$$\min_{U,S,V} J_{LFM} = \|\mathbf{A} - \mathbf{USU}^T\|_F^2 + \|\mathbf{C} - \mathbf{UV}^T\|_F^2$$

où $\|\cdot\|_F$ est la norme de Frobenius et les différents facteurs sont des matrices non négatives. Dans $\|\mathbf{A} - \mathbf{USU}^T\|_F^2$, une factorisation triple est utilisée à la place de la factorisation en deux facteurs classique [LEE 00]. Cette forme symétrique est utilisée pour l'approximation de matrices carrées dont les lignes et les colonnes sont indexées par les mêmes éléments alors que la factorisation habituelle concerne des matrices rectangulaires représentant des relations entre éléments différents. Chaque ligne $\mathbf{U}_i$ dans le facteur $\mathbf{U} \in \mathbb{R}_+^{N \times K}$ correspond à une représentation latente de faible dimension du noeud x_i, qui code conjointement la connectivité et le contenu de ce noeud. Le facteur $\mathbf{S} \in \mathbb{R}_+^{K \times K}$ autorise un degré de liberté supplémentaire permettant à la factorisation triple une meilleure approximation de la matrice d'origine. Les lignes de la matrice $\mathbf{V} \in \mathbb{R}_+^{M \times K}$ représentent la projection de $\mathbf{C}$ sur l'espace latent.

2.2.2. *Techniques de Régularisation de Graphes*

La régularisation dans les graphes est une technique qui a été rendue populaire dans le cadre du développement de modèles semi-supervisés d'apprentissage prenant en compte la proximité entre données [GU 10]. L'idée sous-jacente est que des données proches tendent à avoir la même étiquette. Cette contrainte y est introduite comme un terme de régularisation qui s'ajoute à un coût de classification. Ce type

de régularisation a été utilisée dans un contexte de factorisation de matrices pour le filtrage collaboratif [GU 10]. Soit $\mathbf{W}$ une matrice de ponderation, non négative et symétrique dans un graphe $\mathcal{G}$, soit $\mathbf{u}_i$ le vecteur de caractéristiques associé au noeud x_i. La contrainte de régularisation s'écrit : $\mathcal{R} = \frac{1}{2}\sum_{i,j} \|\mathbf{u}_i - \mathbf{u}_j\|^2 \mathbf{W}_{ij} = tr\big(\mathbf{U}^T(\mathbf{D}-\mathbf{W})\mathbf{U}\big) = tr(\mathbf{U}^T\mathbf{L}\mathbf{U})$, où $tr(\cdot)$ est la trace de la matrice. $\mathbf{L} = \mathbf{D} - \mathbf{W}$ est le laplacien du graphe où $\mathbf{D}$ est une matrice diagonale telle que $\mathbf{D}_{ii} = \sum_j \mathbf{W}_{ij}$.

3. Modèle proposé

Les travaux existants concernant la prédiction de liens utilisent une unique source d'information, qu'elle soit topologique où qu'elle corresponde aux attributs des noeuds. Nous proposons ici d'intégrer trois caractéristiques des réseaux. A notre connaissance, il n'y a pas de travaux similaire dans le cadre de la prédiction temporelle de liens.

3.1. *Structure Temporelle de Liens*

Afin de capturer l'information contenue dans les T états d'une série temporelle de graphes, nous utilisons une ponderation exponentielle des matrices d'adjacence de ces graphes [ACA 09]. Soit $\{A_t(i,j), t = t_0+1, ..., t_0+T\}$ une série de matrices d'adjacence, nous construisons une matrice pondérée de la façon suivante :

$$\mathbf{A}_{(t_0+1)\sim(t_0+T)}(i,j) = \sum_{t=t_0+1}^{t_0+T} \theta^{t_0+T-t} A_t(i,j) \tag{1}$$

où $\theta \in [0,1]$ est la paramètre de lissage. Dans la matrice $\mathbf{A}_{(t_0+1)\sim(t_0+T)}$, l'importance de $\mathbf{A}_t$ décroît d'autant plus que l'index t est grand, i.e., les matrices anciennes ont moins d'importance. $\mathbf{A}_{(t_0+1)\sim(t_0+T)}$que nous noterons $\mathbf{A}$ dans la suite de l'article résume l'évolution temporelle des liens.

3.2. *Spécification du modèle*

3.2.1. *Objectif*

Le modèle proposé combine les deux idées présentées dans les sections 2.2.1 et 2.2.2 dans un modèle unique. Il est basé sur la fonction objective suivante :

$$\begin{aligned}
\min_{U,S,V} J &= \|\mathbf{A} - \mathbf{U}\mathbf{S}\mathbf{U}^T\|_F^2 + \alpha\|\mathbf{C} - \mathbf{U}\mathbf{V}^T\|_F^2 + \lambda tr(\mathbf{U}^T\mathbf{L}\mathbf{U}) \\
&= \sum_{i=1}^{N}\sum_{j=1}^{N}(\mathbf{A}_{ij} - (\mathbf{U}\mathbf{S}\mathbf{U}^T)_{ij})^2 + \alpha\sum_{i=1}^{N}\sum_{j=1}^{M}(\mathbf{C}_{ij} - (\mathbf{U}\mathbf{V}^T)_{ij})^2 \\
&\quad +\lambda tr(\mathbf{U}^T\mathbf{L}\mathbf{U}) \\
&\quad s.t. \quad \mathbf{U} \geq 0, \mathbf{S} \geq 0, \mathbf{V} \geq 0
\end{aligned} \tag{2}$$

où les facteurs matriciels $\mathbf{U} \in \mathbb{R}_+^{N \times K}$, $\mathbf{S} \in \mathbb{R}_+^{K \times K}$, et $\mathbf{V} \in \mathbb{R}_+^{M \times K}$ sont non-négatifs. α et λ sont les paramètres qui permettent de pondérer l'importance de la prise en compte de l'information de contenu et de proximité dans le modèle joint.

Le premier terme $\|\mathbf{A} - \mathbf{USU}^T\|_F^2$ correspond à la factorisation du résumé temporel $\mathbf{A}$. La matrice de liens $\mathbf{A}$ peut être symétrique ou non suivant que le graphe est non dirigé ou dirigé, et la matrice de facteur correspondante $\mathbf{S}$ sera symétrique ou pas. Le second terme $\alpha\|\mathbf{C} - \mathbf{UV}^T\|_F^2$ dans l'équation [2] permet d'incorporer l'information de contenu qui est importante pour la prédiction de liens [KAS 09] [WAN 07].

Le terme de régularisation $\lambda tr(\mathbf{U}^T\mathbf{LU})$ permet d'intégrer l'information de proximité dans le modèle joint, et correspond à une contrainte pour la factorisation de la matrice de liens. Différents types d'information de proximité peuvent être codées dans la matrice $\mathbf{W}$, telle que les mesures de *voisinage commun* où *d'attachement préférentiel* (voir Section 4.1). Les résultats expérimentaux montrent que les caractéristiques topologiques locales sont essentielles pour la prédiction de liens. Sans perte de généralité, nous utilisons ici la mesure de localité basée sur le voisinage commun.

3.2.2. *Inférence*

Après avoir obtenu les facteurs latents $\mathbf{U}$, $\mathbf{S}$ et $\mathbf{V}$, la probabilité d'existence d'un lien peut être obtenue par un produit des facteurs latents $\mathbf{U}$ et $\mathbf{S}$ qui permet d'obtenir la matrice de score suivante :

$$\mathbf{Sc}_{T+1} = f(\mathbf{A}, \mathbf{C}, \mathbf{W}) = \mathbf{USU}^T \tag{3}$$

3.2.3. *Apprentissage*

Afin d'apprendre le modèle proposé, nous utilisons un algorithme d'optimisation alternée [LEE 00] qui garantit la non-négativité des facteurs latents $\mathbf{U}$, $\mathbf{S}$ et $\mathbf{V}$ et permet la sélection automatique du pas d'optimisation. La fonction objectif $J(U, S, V)$ de l'équation [2] n'est pas convexe conjointement pour $\mathbf{U}, \mathbf{S}$ et $\mathbf{V}$. L'algorithme de mise à jour alternée optimise la fonction pour un seul ensemble de paramètres à la fois tandis que les autres sont fixés, puis alterne les paramètres optimisés. Cette procédure est répétée jusqu'à convergence.

Afin d'apprendre la matrice $\mathbf{U}$ tout en fixant les matrices $\mathbf{S}$ et $\mathbf{V}$, il faut résoudre le problème d'optimisation suivant :

$$\min_U J(\mathbf{U}) = \|\mathbf{A} - \mathbf{USU}^T\|_F^2 + \alpha\|\mathbf{C} - \mathbf{UV}^T\|_F^2 + \lambda tr(\mathbf{U}^T\mathbf{LU}) \tag{4}$$

La dérivée de $J(\mathbf{U})$ par rapport à $\mathbf{U}$ s'écrit :

$$\begin{aligned} \frac{\partial J(\mathbf{U})}{\partial \mathbf{U}} &= 2(\mathbf{USU}^T\mathbf{US}^T + \mathbf{US}^T\mathbf{U}^T\mathbf{US} - \mathbf{AUS}^T - \mathbf{A}^T\mathbf{US}) \\ &+ 2\alpha(\mathbf{UV}^T\mathbf{V} - \mathbf{CV}) + 2\lambda\mathbf{LU} \end{aligned} \tag{5}$$

En utilisant la condition de Karush-Kuhn-Tucker pour la non-négativité de $\mathbf{U}$ ainsi que le fait que $\frac{\partial J(\mathbf{U})}{\partial \mathbf{U}} = 0$, nous obtenons la règle de mise à jour suivante :

$$\mathbf{U}_{ij} \longleftarrow \mathbf{U}_{ij} \sqrt{\frac{[\mathbf{AUS}^T + \mathbf{A}^T\mathbf{US} + \alpha\mathbf{CV} + \lambda\mathbf{WU}]_{ij}}{[\mathbf{USU}^T\mathbf{US}^T + \mathbf{US}^T\mathbf{U}^T\mathbf{US} + \alpha\mathbf{UV}^T\mathbf{V} + \lambda\mathbf{DU}]_{ij}}} \tag{6}$$

Les matrices $\mathbf{S}$ et $\mathbf{V}$ peuvent être apprises de la même manière. Les règles de mise à jour obtenues pour $\mathbf{S}$ et $\mathbf{V}$ sont respectivement :

$$\mathbf{S}_{ij} \longleftarrow \mathbf{S}_{ij} \sqrt{\frac{[\mathbf{U}^T\mathbf{AU}]_{ij}}{[\mathbf{U}^T\mathbf{USU}^T\mathbf{U}]_{ij}}} \quad \mathbf{V}_{ij} \longleftarrow \mathbf{V}_{ij} \frac{[\mathbf{C}^T\mathbf{U}]_{ij}}{[\mathbf{VU}^T\mathbf{U}]_{ij}} \tag{7}$$

3.2.4. *Analyse de la Convergence*

A partir des règles de mise à jour pour les facteurs $\mathbf{U}$ - équation [6] -, $\mathbf{S}$ - équation [7] -, et $\mathbf{V}$ - équation [7] -, nous pouvons prouver que l'algorithme d'apprentissage converge.

Theorem 1 *En utilisant de manière alternée les règles de mise à jour précédentes, la fonction objectif définie dans l'équation [2] décroît de manière monotone et donc l'algorithme d'apprentissage converge vers un optimum local.*

La preuve est omise à cause du nombre de pages limité. [2]

3.2.5. *Analyse de la complexité*

La complexité pour mettre à jour $\mathbf{U}$ à chaque itération est $O(NK^2)$, où N est le nombre de noeuds du réseau et K la dimension du facteur U. De même, la complexité est respectivement de $O(NK^2)$ et $O(NK)$ pour la mise à jour de $\mathbf{S}$ et $\mathbf{V}$. Etant donné que $K \ll N$, la complexité finale de l'algorithme est $O(N)$ ce qui permet la prise en charge de grandes masses de données.

4. Expériences

4.1. *Protocole Expérimental*

Dans les expériences, nous examinons comment notre modèle se comporte sur des réseaux dynamiques issus du monde réel. Nous considérons ici 4 réseaux bibliographiques extraits de l'archive arXiv allant des années 1992 aux années 2002 [LIB 03]. Ces différents réseaux proviennent de différentes sections de arXiv : *Condensed Matter (Cond-mat), General Relativity et Quantum Cosmology (Gr-qc), High energy physics phenomenology (Hep-ph)* et *High energy physics theory (Hep-th).*

2. Elle est disponible sur le site des auteurs de l'article.

Nous introduisons au préalable les différents modèles de base utilisés pour la comparaison avec notre approche :

Modèle **Dependent Prediction** : c'est une méthode naïve qui utilise la matrice de contingence la plus récente A_{t_0+T} comme score prédit : $\mathbf{Sc}_{T+1} = \mathbf{A}_{t_0+T}$.

Modèle **Weighted Dependent Prediction** : Cette méthode utilise le graphe pondéré $\mathbf{A}$ comme matrice de score prédit : $\mathbf{Sc}_{T+1} = \mathbf{A}$.

Modèle **Common Neighbor** : Soit $\mathcal{G}^{'}$ le graphe d'ensemble de sommets où deux noeuds sont connectés si $\mathbf{A}_{ij} > 0$, le score prédit par ce modèle pour la création d'un lien (i, j) correspond au nombre de voisins commun aux deux noeuds dans le graphe $\mathcal{G}^{'}$: $[\mathbf{Sc}_{T+1}]_{ij} = |\Gamma(i) \cap \Gamma(j)|$, où $\Gamma(i)$ est l'ensemble de voisins du noeud i dans $\mathcal{G}^{'}$.

La méthode d'attachement préférentiel (**Preferential Attachment** [BAR 99]) utilise le produit du degré des noeuds de $\mathcal{G}^{'}$ en tant que matrice de score : $[\mathbf{Sc}_{T+1}]_{ij} = |\Gamma(i)| \cdot |\Gamma(j)|$.

La méthode de **Katz** [LIB 03] consiste à additionner tous les chemins entre deux noeuds de $\mathcal{G}^{'}$ en les pondérant exponentiellement en fonction de leur longueur afin de donner plus d'importance aux chemins courts : $\mathbf{Sc}_{T+1} = \sum_{l=1}^{\infty} \beta^l \cdot |path_{i,j}^{<l>}|$, où $path_{i,j}^{<l>}$ est l'ensemble des chemins de longueur l entre les noeuds i et j.

La méthode **NMF** [LIB 03] appliquée directement à la matrice $\mathbf{A}$ qui permet de calculer un score uniquement en fonction de la structure du réseau.

La méthode **GNMF** [CAI 08] qui combine la proximité et la structure du réseau.

La méthode **GRJMF** (*Graph Regularized Joint Matrix Factorization*) qui correspond à notre méthode.

La taille de la fenêtre temporelle glissante pour l'apprentissage du modèle a été fixée à $T = 3$[3]. Etant donné cette taille $T = 3$ et le fait que les corpus commencent en 1992, nous évaluons nos modèles sur les années 1995 à 2002. Afin d'évaluer les performances des différentes approches, nous utilisons la mesure AUC (score Area Under the receiver operating characteristic Curve), qui est une mesure robuste utilisée classiquement [ACA 09]. Nous avons évalué notre méthodes en moyennant les performances obtenues sur 5 découpages différents des données. La matrice de similarité a été calculée par la méthode du *voisinage commun*.

4.2. *Résultats de la prédiction temporelle de liens*

Nous avons comparé les performances des différentes approches sur les 4 jeux de données. Les paramètres des différentes méthodes ont été réglés manuellement et nous donnons ici les meilleurs résultats obtenus pour chaque méthode avec une

3. Des tailles différentes permettent d'obtenir des performances différentes ; $T = 3$ correspond à la taille qui obtient les meilleures performances sur les 4 jeu de données.

	DP	WP	CN	PA	Katz	NMF	GNMF	GRJMF
Condm	0.6585	0.7068	0.6757	0.7152	0.7231	0.7195	*0.7261*	**0.7605**
Gr-qc	0.6288	0.6702	0.6375	0.6707	*0.7047*	0.6680	0.6756	**0.7588**
Hep-ph	0.6103	0.6549	0.6328	0.7112	*0.7649*	0.7323	0.7437	**0.7896**
Hep-th	0.6454	0.6911	0.6635	0.7216	*0.7518*	0.7133	0.7274	**0.7977**

Tableau 1. *AUC moyen des différentes méthodes, moyenné sur les années entre 1995 et 2002, pour une fenêtre de taille $T = 3$ et pour chaque corpus. Les meilleures performances illustrées en gras, les secondes meilleures en italique.*

Year	DP	WP	CN	PA	Katz	NMF	GNMF	GRJMF
1995	0.6306	0.6621	0.6490	0.7061	*0.7141*	0.6612	0.6683	**0.7186**
1996	0.6631	0.6962	0.6764	0.7028	*0.7122*	0.7094	0.7097	**0.7460**
1997	0.6086	0.6993	0.6451	0.7187	*0.7115*	0.6869	0.6907	**0.7374**
1998	0.6214	0.7133	0.6872	0.7074	0.7137	0.7293	*0.7360*	**0.7628**
1999	0.6864	0.7154	0.7142	0.7111	0.7330	0.7393	*0.7365*	**0.7979**
2000	0.6328	0.6987	0.6785	0.7004	0.7198	0.7083	*0.7284*	**0.7756**
2001	0.6870	0.7376	0.7177	0.7195	*0.7466*	0.7485	0.7403	**0.7876**
2002	0.7026	0.7322	0.7349	0.7277	0.7580	0.7466	*0.7598*	**0.8083**
Ave.	0.6585	0.7068	0.6757	0.7152	0.7231	0.7195	0.7261	**0.7605**

Tableau 2. *AUC moyen sur le corpus Cond-mat pour chaque année de test. Les meilleures performances sont en gras, les secondes meilleures en italique.*

combinaison optimale de paramètres. La dimension des facteurs latents (variable K) a aussi été déterminée expérimentalement et les résultats donnés correspondent aux valeurs suivantes : $K = 60$, $\alpha = 1$, $\theta = 0.4$ et $\lambda = 1$. Le nombre d'itérations de l'algorithme d'apprentissage a été fixé à 100. L'influence des différents paramètres est étudiée dans la section suivante.

Les résultats sont illustrés en Table 1 pour la tâche de prédiction temporelle de liens. On peut observer que la méthode la plus performante est la méthode GRJMF qui correspond au modèle proposé dans cet article. Cela indique que notre approche est capable d'intégrer efficacement les différentes sources d'information : la structure du graphe, le contenu des noeuds et l'information de proximité. La Table 2 présente les performances année par année sur le corpus *Cond-mat*. Parmi les modèles qui utilisent l'information de proximité (CN, PA, Katz), la méthode de Katz obtient les meilleures performances et est souvent meilleure que les modèles à base de factorisation non-négative (NMF ou GNMF) ce qui tend à prouver que la proximité est une information particulièrement pertinente pour la prédiction de liens temporels.

Nous évaluons aussi la contribution des différentes sources d'informations (information de proximité, contenu et structure) sur les performances de notre méthode. La Figure 2 présente les résultats obtenus pour différentes variantes de notre méthode. GRJMF-L correspond au modèle basé sur la structure uniquement ($\alpha = 0$ et $\lambda = 0$) ; GRJMF-L+C correspond au modèle qui utilise conjointement l'information de struc-

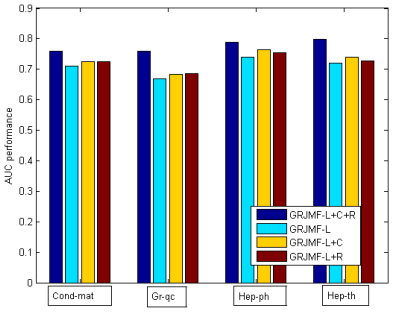


Figure 2. *AUC moyen pour les différentes variantes du modèle GRJMF.*

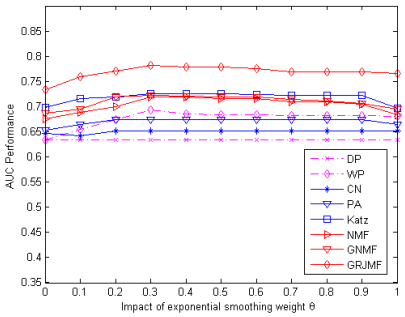


Figure 3. *Impact de θ sur le corpus Gr-qc.*

ture et de contenu ($\lambda = 0$) ; GRJMF-L+R est le modèle qui utilise le terme de régularisation mais pas le contenu des noeuds ($\alpha = 0$) ; enfin GRJMF-L+C+R correspond au modèle complet. On peut constater que le modèle GRJMF-L est le modèle le moins performant tandis que l'utilisation conjointe de la structure et du contenu ou bien encore de la structure et de l'information de proximité permettent d'obtenir des performances sensiblement équivalentes. L'intégration simultanée de toutes les sources d'information - modèle GRJMF-L+C+R - améliore les résultats par rapport aux différentes variantes du modèle.

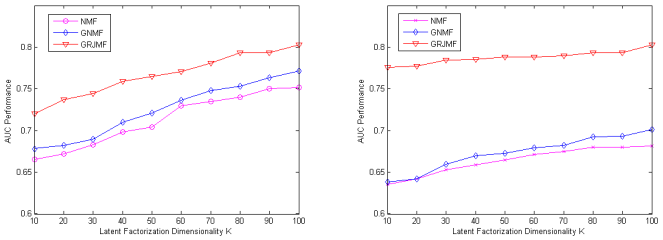
4.3. *Impact de la valeur des paramètres*

Nous étudions ici l'impact de la valeur des différents paramètres (θ, K, α et λ) du modèle. Pour cela, nous avons effectué un ensemble d'expériences avec des valeurs différentes de ces paramètres.

Impact de θ : Nous avons utilisé le corpus *Gr-qc* afin de vérifier l'impact de différentes valeurs de θ (Figure 3). Les résultats obtenus sur les trois autres jeux de données sont identiques. Nous pouvons constater que la valeur de θ optimale est ici $\theta = 0.4$ et que des valeurs inférieures et supérieures dégradent les résultats.

Impact de K : La Figure 4 montre les performances obtenues pour différentes tailles de fenêtre temporelle, pour K variant de 10 à 100. On constate que plus K augmente, et plus les performances sont bonnes. Afin de garder un bon compromis entre performance et rapidité, nous avons fixé K à 60 dans les expériences.

Impact de α : Nous avons fait varier la valeur de α dans l'intervalle $[0, 2]$ afin d'examiner l'impact de l'information de contenu. Les résultats sont reportés Figure 5 pour le corpus *Gr-qc*. $\alpha = 0$ correspond au modèle GNMF (pas de contenu). On peut constater que l'information de contenu est une information importante étant donné que la performance du modèle s'accroît rapidement dès que α est non nul. La per-



(a) Cond-mat dataset (b) Gr-qc dataset

(c) Hep-ph dataset (d) Hep-th dataset

Figure 4. *AUC moyen en fonction de la taille de la fenêtre temporelle (paramètre K).*

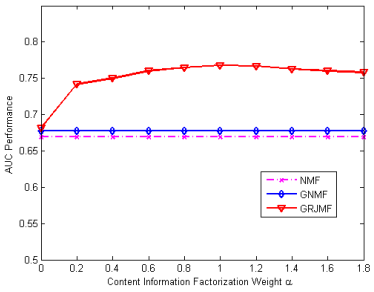


Figure 5. *Impact de* α *.*

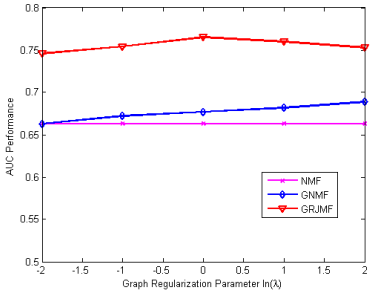


Figure 6. *Impact de* λ*.*

formance la meilleure est obtenue pour $\alpha = 1$, des valeurs supérieures donnant trop d'importance au contenu au détriment des autres sources d'information (structure et proximité).

Impact de λ **:** nous avons fait varier λ pour les valeurs $\{0.01, 0.1, 1, 10, 100\}$. La Figure 6 reporte les résultats obtenus pour le corpus *Gr-qc*. Nous pouvons constater que λ a une influence sur la performance du modèle, les performances sont meilleures avec $\lambda \neq 0$. Cela confirme qu'il est important de considérer l'information de proximité pour la prédiction de liens temporels.

5. Conclusion

Dans ce papier, nous avons développé un nouveau modèle pour la prédiction de liens temporels dans les réseaux sociaux dynamiques. Notre approche permet la prise en compte unifiée de plusieurs sources d'information : contenu, structure et proximité. Le modèle est basé sur des techniques de factorisation matricielle avec des matrices non négatives et sur l'utilisation de la régularisation dans les graphes. Nous avons utilisé une méthode d'optimisation alternée permettant l'apprentissage de ces modèles sur des données réelles. Les expériences montrent l'efficacité de notre méthode en comparaison de l'état de l'art et permettent de mieux comprendre l'utilité des différentes sources d'information utilisées dans notre modèle.

6. Remerciements

Ce travail a été partiellement financé par l'Agence Nationale pour la Recherche (Projet Fragrances, ANR-08-CORD-008-01).

7. Bibliographie

[ACA 09] ACAR E., DUNLAVY D. M., KOLDA T. G., « Link Prediction on Evolving Data Using Matrix and Tensor Factorizations », *ICDM Workshops*, , 2009, p. 262-269.

[BAR 99] BARABASI A. L., ALBERT R., « Emergence of scaling in random networks », *Science*, vol. 286, 1999, p. 509-512.

[CAI 08] CAI D., HE X., WU X., HAN J., « Non-negative matrix factorization on manifold », *ICDM*, , 2008, p. 63-72.

[GET 05] GETOOR L., DIEHL C. P., « Link mining : a survey », *ACM SIGKDD Explorations Newslett.*, vol. 7, 2005, p. 3-12.

[GU 10] GU Q., ZHOU J., DING C., « Collaborative Filtering : Weighted Nonnegative Matrix Factorization Incorporating User and Item Graphs », *SDM*, , 2010, p. 199-210.

[HUA 09] HUANG Z., LIN D. K. J., « The Time-Series Link Prediction Problem with Applications in Communication Surveillance », *INFORMS JOURNAL ON COMPUTING*, , 2009, p. 286-303.

[KAS 09] KASHIMA H., KATO T., YAMANISHI Y., SUGIYAMA M., TSUDA K., « Link Propagation : A Fast Semi-supervised Learning Algorithm for Link Prediction », *SDM*, , 2009, p. 1099-1110.

[LEE 00] LEE D. D., SEUNG H. S., « Algorithms for non-negative matrix factorization », *NIPS*, , 2000, p. 556-562.

[LIB 03] LIBEN-NOWELL D., KLEINBERG J., « The link prediction problem for social networks », *CIKM*, , 2003, p. 556-559.

[WAN 07] WANG C., SATULURI V., PARTHASARATHY S., « Local Probabilistic Models for Link Prediction », *ICDM*, , 2007, p. 322-331.

[ZHU 07] ZHU S., YU K., CHI Y., GONG Y., « Combining content and link for classification using matrix factorization », *SIGIR*, , 2007, p. 487-494.