Analyse et application de la diffusion d'information dans les microblogs

Dong Wang

Communauté
UNIVERSITÉ Grenoble Alpes

中国科学院计算技术研究所
Institute of Computing Technology, Chinese Academy of Sciences

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

préparée dans le cadre d'une cotutelle entre l'*Université Grenoble Alpes* et l'académie des sciences de Chine

Spécialité : **Informatique**

Arrêté ministériel : le 6 janvier 2005 - 7 août 2006

Analyse et application de la diffusion d'information dans les micro-blogs

Présentée par

« Dong /WANG»

Thèse dirigée par **« Mohamed Ali KAFAAR »**, **«Kavé SALAMATIAN »** et **« Gaogang Xie »**

préparée au sein des **Laboratoire LISTIC, INRIA Grenoble Alpes et l'Institut des Technologies de l'Information de l'académie des Sciences de Chine**

dans **les Écoles Doctorales SISEO (Sciences et Ingénierie des Systèmes, de l'Environnement et des Organisations) et de l'académie des Sciences de Chine**

Thèse soutenue publiquement le **« 22 octobre 2015 »**,
devant le jury composé de :

Stéphane, GRUMBACH
Directeur de Recherche à l'INRIA, Président et Rapporteur
Sue, MOON
Professeur à KAIST, Corée du SUD, Rapporteur
Hamed, HADDADI
Chargé de Cours à l'université Queen Mary de Londres, Membre
Fehmi, BEN ABDESSALLEM
Chercheur au CISC, Suède, Membre
Aurélien, FARAVELON
Chercheur à l'ENS Lyon, Invité

The Analysis and Applications of Information Diffusion in Microblogs

by

Dong Wang

Submitted to the Université de Grenoble
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

UNIVERSITÉ DE GRENOBLE

October 2015

Author .....
Université de Grenoble
October 20, 2015

Certified by .....
Kavé Salamatian, Mohamed-Ali Kaafar and Gaogang Xie
Professor
Thesis Supervisor

Accepted by .....
XXXX
Chairman, Department Committee on Graduate Theses

1-Introduction

Avec Le développement du Web 2.0, les services de micro-blogging (comme Twitter et Sina Weibo) sont devenus les plateformes essentielles pour la diffusion d'information en ligne. Ces plateformes permettent les aux utilisateurs de suivre d'autres utilisateurs et recevoir les informations qu'ils diffusent. Ces informations sont envoyées sous forme de messages courts avec une longueur limitée, à 140 caractères par exemple pour Twitter ou Sina Weibo, qui sont appelés des « *tweet* ». Les personnes recevant des tweets peuvent les retransmettre, les *retweeter* à leur propre réseau de suiveurs. Ce processus de diffusion de l'information qui est bien différent de l'approche classique des médias en ligne ou hors ligne, a un impact important sur les habitudes de consommation de contenus des internautes. Ainsi, le mécanisme de *retweet* accélère la diffusion de l'information

Le tremblement de terre dramatique de Mars 2011 illustre très bien ce phénomène d'accélération de la propagation d'information. Je présente dans la Figure 1, le volume de trafic sur Twitter dans l'heure suivant le tremblement de terre. Les lignes de couleurs sont les tweets publiés au Japon et les lignes

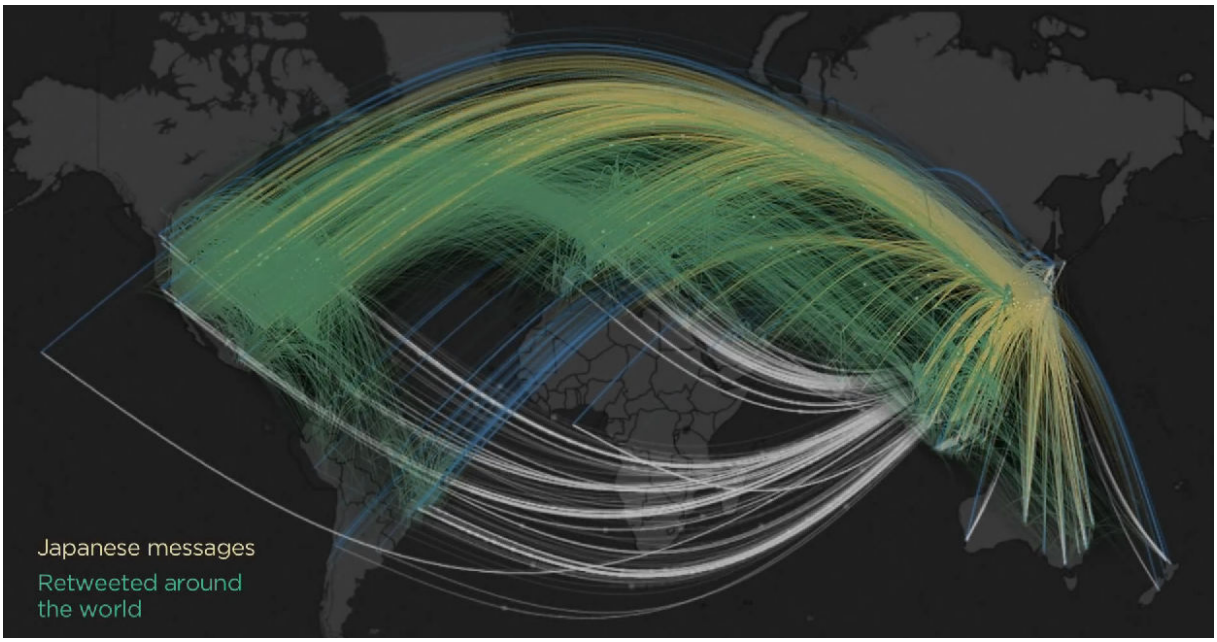


Figure 1- Le trafic sur Twitter durant la première heure après le tremblement de terre de mars 2011

blanches sont les réponses à ceux-ci. Juste dans la première heure du tremblement de terre, plus de 5000 tweets par minutes seront générés.

L'observation des tendances des mots-clés ainsi que de la popularité des recherches sur les outils de recherche sur cette même période montre la montée brusque et rapide concomitante de mot clé « *Japan Earthquake* » dans ces deux systèmes de diffusion de l'information (voir Figure 2)

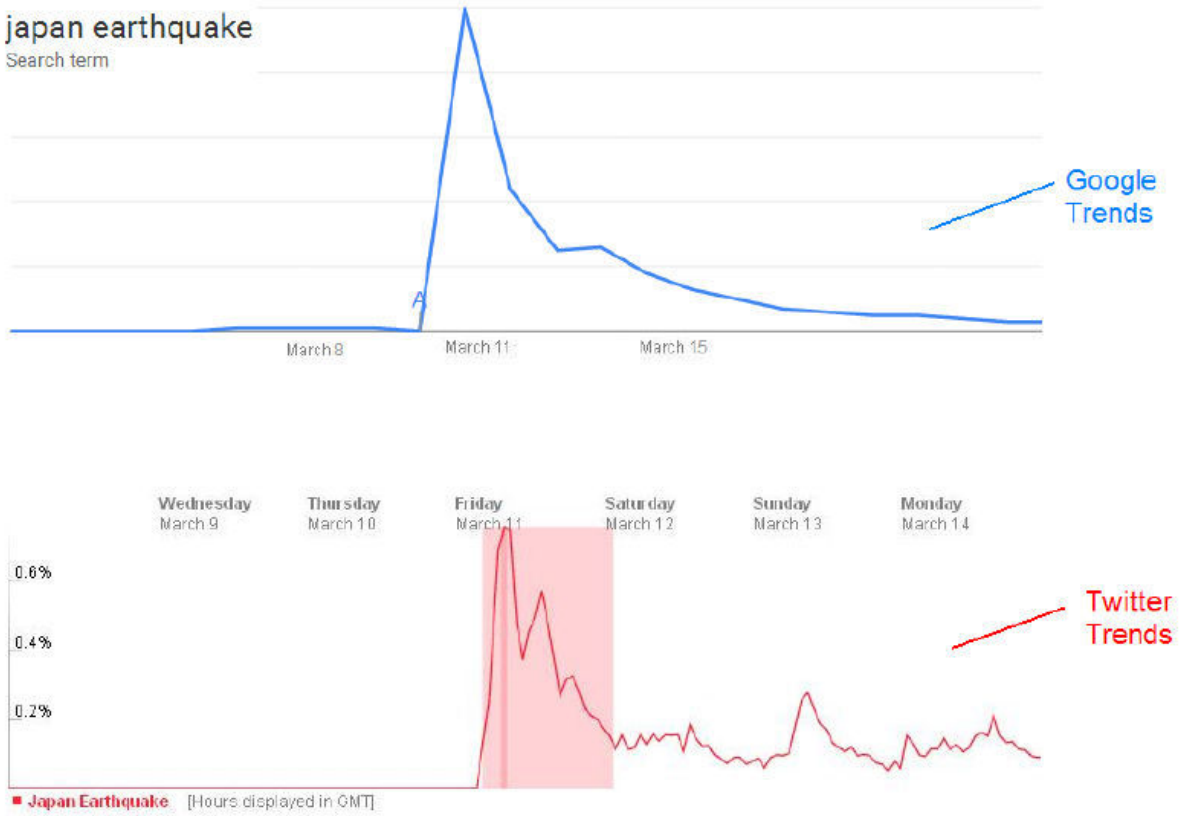


Figure 2-Les tendances sur le mot clé « Japan Earthquake » durant le mois de mars 2011 dans Google et sur Twitter

Ainsi, le suivi et l'analyse de la diffusion d'information sur les plateformes de microblogging est un outil essentiel dans l'extraction de l'opinion publique, pour les systèmes de recommandation et pour l'optimisation de la recherche d'information. Ceci est l'objectif principal de cette thèse.

1.1-*Principaux défis*

L'analyse de la diffusion d'information sur les médias sociaux consiste en trois étapes. Dans une première étape, il convient de récupérer un ensemble de données représentatif et sans biais de services de microblog. Dans une seconde phase, cet ensemble de données est utilisé par le chercheur dont les observations aboutissent au développement d'un modèle, éventuellement analytique de la diffusion d'information. Finalement ce modèle est utilisé afin de créer une application, ce qui finalement l'objectif ultime de l'analyse.

Chacune des trois étapes citées plus haut sont la source de défis considérables que je décrirais dans la suite :

1. L'échantillonnage des données et plus particulièrement des graphes est complexe. Les services de micro-blog attirent un nombre croissant d'utilisateur. En Février 2013 il y'avait plus de 200 millions d'utilisateurs actifs sur Twitter et Sina Weibo a rapporté plis de 300 millions d'utilisateurs en Mai 2012. Traiter de tels volumes d'utilisateurs et le torrent continu d'information qu'ils diffusent est impossible pour le chercheur. De plus les sociétés gérant les services de micro-blog sont très rétives à l'idée de partager les données relatives à l'utilisation de leurs services. Ceci est dû à de nombreux facteurs. D'une part les micro-blogs contiennent un volume très important d'information à caractère personnel qui ne peuvent se retrouver dans la nature sans protection. D'autre part, le la diffusion d'information est dans le cœur de la principale source de revenus des services de micro-blog, c'est à dire la publicité en ligne. Ainsi les informations de ces services sont des données sensibles au niveau de leurs activités économiques et il y'a un risque de compétition accru à partager ces données. En résumé même si le volume d'information était gérable et traitable, les considérations relatives au respect des

informations privées ainsi que le caractère sensible au niveau économique des données d'utilisation des micro-blogs, nous obligent à toujours considérer que les données que nous avons ne sont qu'un échantillon de la globalité de l'activité du service de micro-blog et qu'ils convient donc d'y appliquer des précautions statistiques classiques comme la représentativité, le biais, *etc.*

2. L'analyse et la modélisation du processus de propagation est aussi complexe. En effet la diffusion d'une information dépend de son sujet, de sa qualité, de la topologie des réseaux de diffusions, des caractéristiques sociales des utilisateurs. Certains de ces éléments sont difficile à intégrer au sein d'un modèle unique, consistant et cohérent. De plus l'étude mathématique de ces modèles peut être complexe et trouver des résultats analytiques qui puissent être utilisé pour dimensionner les réseaux sociaux ou interpréter les observations empiriques est un défi à lui seul. En effet afin de prendre en compte la complexité des interactions dans ces réseaux il convient de s'éloigner des hypothèses classiques d'indépendances et d'ajouter des corrélations ou des mécanismes ce qui rend les modèles beaucoup plus complexes.

3. L'application et l'utilisation des résultats afin d'aboutir à des usages concrets est le dernier défi que nous avons à relever dans le cadre de cette thèse. En effet, il existe dans la littérature beaucoup de travaux mettant en évidence les interactions entre les réseaux sociaux et l'information, l'intérêt des internautes ou la popularité des contenus en ligne. Mais la transformation de cette relation en un application concrète peut être difficile compte tenu du volume d'information à traiter, des contraintes de temps réel (ou semi-réel). Ces relations apparaissent fréquemment comme des signaux faibles qu'il convient d'amplifier en combinant plusieurs sources d'informations ce qui requiert des

méthodologies de fusion d'informations hétérogènes qui sont parfois difficile à manipuler.

1.2- Contributions

Cette thèse présente une analyse globale de la diffusion d'information dans les micro-blogs et plus généralement dans les réseaux sociaux. Elle tente de répondre à certains des trois défis présentés plus haut. Les contributions et innovations présentés dans cette thèse sont les suivantes :

1)- Les deux approches les plus populaires d'échantillonnage de réseaux sociaux sont la marche aléatoire de Hasting Metropolis, *Metropolis-Hastings Random Walk (MHRW)*, et l'échantillonnage sans biais de graphes dirigés, *Unbiased Sampling in Directed Social Graph* (*USDSG*). Quand ces deux approches sont appliquées aux réseaux sociaux et en particulier à ceux fondés sur les micro-blogs, ils génèrent un nombre considérable d'auto-échantillonnage, *i.e.*, de ré-échantillonnage de nœuds déjà choisi. Ceci réduit fortement l'efficacité de ces mécanismes et la qualité de l'échantillonnage. Afin de réduire ce problème j'ai développé un modèle de l'échantillonnage sur les graphes sous forme de chaîne de Markov et j'en ai déduit les conditions nécessaires et suffisantes garantissant un échantillonnage sans biais. Sur la base de ces conditions j'ai proposé un nouvel algorithme efficace et sans biais qui réduit la probabilité d'auto-échantillonnage des approches MHRW et USDSG en distribuant uniformément cette probabilité sur les probabilités de transition d'un nœud à l'autre tout en gardant la propriété sans-biais. Ce nouveau schéma d'échantillonnage est appelé échantillonnage sans biais avec nœuds factice, *Unbiased Sampling with Dummy Edges* (USDE). L'évaluation montre qu'alors que le degré moyen des nœuds échantillonné par MHRW et USDSG est 2 à 4 fois plus élevé que le graphe initial, USDE atteint un degré moyen très proche tout en évitant les répétitions dans l'échantillonnage. De plus le temps d'échantillonnage moyen par nœuds pour USDE est de la moitié de celui que nécessite MHRW ou USDSG. Ceci valide l'intérêt d'USDE pour l'échantillonnage sans biais des graphes de réseaux sociaux.

2) la seconde contribution de cette thèse vise la modélisation de la diffusion d'information dans les réseaux de micro-blogs. Les principaux modèles utilisé dans littérature sont le modèle de cascades indépendantes (*Independent Cascade Model*) et le modèle de seuil linéaire (*Linear Threshold Model*). Ces deux modèles ne prennent pas en compte l'évolution de l'intérêt d'un message diffusé sur le réseau social. Ils ne permettent donc pas d'obtenir une bonne prédiction de la portée de la diffusion des informations. J'ai développé un nouveau modèle fondé sur le modèle classique des arbres de Galton-Watson mais qui est modifié afin de prendre en compte l'éphéméralité des messages. Ainsi ce modèle prend en compte les trois propriétés importantes expliquant la diffusion de l'information dans les réseaux sociaux : l'intérêt intrinsèque de l'information et son éphéméralité, la topologie du réseau social et les propriétés de la source d'information et des retwetteurs. J'ai validé ce nouveau modèle sur des jeux de données issus de Sina Weibo et de Twitter. J'ai observé que ce nouveau modèle peut prédire de façon fiable dans plus de 82% des cas l'audience d'un message, le nombre de récepteurs qui ont reçu l'information, et dans 90% des cas le nombre maximum de retweet dans les arbres de diffusions issue des micro-blogs. De plus ce modèle permet d'extraire les facteurs endogènes et exogènes qui affectent la popularité des tweets.

3) La troisième contribution de cette thèse pose la question de la relation entre la popularité d'un sujet sur les micro-blogs et sur le web. En effet alors que les micro-blogs agissent principalement comme des médias de communication permettant la diffusion à large échelle d'une information partielle et condensée, le web permet d'approfondir cette information, ce qui devrait se traduire dans le volume de recherche relatif à celle-ci. De nombreux travaux visent à utiliser pour diverses applications cette interaction entre ces deux composantes essentielles du cyberespace, *eg*, la prédiction de cours d'actions sur les marchés boursiers. Néanmoins peu d'études complètes ont été mené permettant d'évaluer cette interaction et ainsi la plupart des travaux publiés dans la littérature apparaissent comme anecdotiques et opportunistes. J'ai dans ma thèse effectué une étude approfondie de la corrélation entre la

popularité des sujets dans les micro-blogs avec les recherches effectuées sur le Web. J'ai montré de façon empirique que les tendances dans les micro-blogs et sur le web partagent aussi bien au niveau temporel qu'au niveau spatial des caractéristiques temporelles. J'ai aussi montré que la croissance d'intérêt sur un sujet dans les micro-blogs peut précéder de quelques heures celle sur le web. Néanmoins la popularité d'un sujet sur les micro-blogs affiche un plus grand niveau de variabilité comparée au web. Ainsi des sujets peuvent émerger très rapidement sur les micro-blogs et perdre leur intérêt aussi rapidement, tandis que l'inertie dans le web est plus importante. Cette analyse ouvre la voie pour la quatrième contribution de cette thèse.

4) Je développe finalement une application de l'analyse précédemment présenté, au marketing par le biais des outils de recherche, *Search Engine Marketing* (*SEM*). En particulier j'ai développé une analyse économique du marché d'intermédiaire en SEM. Ces intermédiaires répondent à la demande des clients qui souhaitent attirer des internautes sur leurs pages web et sont rémunérés pour ceci. A chaque fois qu'une recherche contenant un de ces mots-clés est demandé, le site de recherche met aux enchères les emplacements publicitaires aux différents acteurs ayant soumis une demande et choisi l'annonce publicitaire qui lui garantit le meilleur revenu. Le rôle de l'intermédiaire consiste donc à trouver pour son client un ensemble de mots-clés à même de gagner suffisamment d'enchères tout en dépassant pas un budget donné. L'analyse économique montre l'importance pour le revenu de l'intermédiaire et pour le viabilité d'une campagne publicitaire du prix par clic moyen. L'intermédiaire doit donc trouver des mots-clés qui n'ont pas encore attiré l'attention des compétiteurs et dont le prix d'enchères n'est pas encore élevé, mais qui ont un potentiel important d'attirer l'attention et d'avoir un volume de recherche important. Le fait que la popularité sur les micro-blogs devance de quelques jours la croissance de l'intérêt sur les sites de recherche apporte une observation intéressante pour les SEMs. Le fait que des mots-clés peuvent être détectés sur les micro-blogs avant qu'ils ne deviennent détectable par le volume de recherche web, rend possible l'utilisation de ceux-ci comme candidats potentiels pour le SEM. Afin d'évaluer l'utilisation de ces mots-clés,

j'ai développé une méthodologie de constitution et d'évaluation de portefeuilles de mots-clés pour le SEM. Cette méthodologie inspirée des techniques d'optimisation de portefeuilles d'actions boursières, formalise la balance entre profitabilité escomptée d'un portefeuille de mots-clés, et le risque de ne pas atteindre les objectifs de la campagne publicitaire ou d'avoir une perte au lieu d'un bénéfice réel. J'utilise cette méthodologie pour construire des portefeuilles augmentés avec des mots-clés issues de sujets populaires sur les micro-blogs et je compare la performance de ces portefeuilles avec celle des portefeuilles constitué grâce au techniques courantes. Les résultats montrent un bénéfice moyen quatre fois supérieurs à un même niveau de risque pour les portefeuilles augmentés que pour les portefeuilles classiques. Cette application, de la méthodologie de constitution de portefeuilles et d'évaluation de sa performance, montre aussi sa pertinence pour le SEM.

2-Echantillonnage sans biais des réseaux sociaux

2.1- Introduction

La grande taille des réseaux sociaux rend fréquemment impossible une observation exhaustive de ceux-ci. L'alternative consiste à obtenir un échantillon représentatif du réseau. Cet échantillon doit être représentatif au sens que les paramètres statistiques du réseau initial (non échantillonné) doivent pouvoir être estimé correctement (sans biais) et avec une erreur relativement faible sur l'échantillon obtenu (consistance). L'échantillonnage des graphes suit généralement deux approches. Dans une première approche on vise l'échantillonnage de liens [77,75], *i.e.*, les liens du graphe sont échantillonnés uniformément. La seconde approche vise l'échantillonnage de nœuds [32,91,101], *i.e.*, des nœuds du graphe sont échantillonnés uniformément. Etant donné que les réseaux sociaux sont plus orientés vers les utilisateurs, je ne considère dans cette thèse que la seconde approche.

La qualité d'un échantillon dépend *in fine* de l'application visée. Ainsi par exemple si le but principal est l'estimation des propriétés des nœuds du graphe comme la distribution de degré, il n'est pas nécessaire que le graphe échantillonné soit connexe [78,79], par contre si nous souhaitons étudier des propriétés relatives à la diffusion d'information dans un graphe la connexité du graphe échantillonné est primordiale [85,90]. Ainsi une partie importante de la littérature sur l'échantillonnage des graphes c'est concentré sur des méthodes permettant simultanément d'assurer un échantillonnage sans biais et d'obtenir un graphe échantillonné connexe.

Une de ces méthodes les plus citées est la marche aléatoire de Métropolis-Hasting (MHRW) [32] et ces variantes [33,72,101]. Cette méthode a été initialement conçue pour le réseau social Facebook ou les nœuds se connectent plus fréquemment avec d'autres nœuds ayant un nombre similaire de voisins. L'intuition derrière la méthode MHRW est d'éviter de choisir des nœuds avec un degré très élevé en augmentant explicitement la probabilité de choisir un nœud de degré faible durant le processus d'échantillonnage.

L'application de l'approche MHRW sur les micro-blogs, *e.g.*, Twitter, fait apparaître deux limitations principales. *Primo*, les utilisateurs des micro-blogs tendent à suivre certains utilisateurs très populaires avec des degrés de connectivité qui sont plus ordre de grandeur plus grand [65,66]. Ainsi les nœuds populaires sont connectés à un grand nombre de nœuds de faibles degrés. Le mécanisme MHRW sera ainsi rapidement coincé pour de longue durée de temps dans des nœuds à faible degré et incapable de trouver de nouveaux nœuds. La seconde limitation est liée à la répétition de nœud de faible degré dans l'échantillonnage. Alors que ces nœuds ré-échantillonnés peuvent être réutilisés dans l'estimation des propriétés topologiques, un nœud échantillonné ne comptera qu'une fois dans le contexte de l'étude du comportement individuel d'utilisateurs, même si ce nœud a été choisi plusieurs fois. En supprimant les répétitions, l'estimation donnée par MHRW n'est plus sans biais [33,72,79].

Les deux limitations précédentes sont la principale motivation de mes travaux sur l'échantillonnage sans biais des graphes.

2.2- *Modélisation de la marche aléatoire pour l'échantillonnage*

Je m'intéresse aux méthodes où l'échantillonnage commence en un point initial jouant le rôle de graine. A chaque étape de l'échantillonnage tous les voisins d'un nœud échantillonné sont des candidats potentiels pour l'étape suivante. L'échantillonnage se poursuit en suivant un lien des liens suivant une règle prédéterminée et en arrivant au nouveau nœud qui peut être considéré comme candidat à l'échantillonnage. Un algorithme d'échantillonnage peut être considéré comme sans biais si la probabilité de visiter chacun des nœuds durant le processus est uniforme. De plus, il est souhaitable que le graphe échantillonné soit connecté.

La probabilité de choisir un nœud à l'étape T+1 sachant le nœud échantillonné à l'instant T ne dépend que de ce dernier nœud et non pas de la séquence des nœuds, ce qui signifie que le processus d'échantillonnage peut être représenté par une chaîne de Markov dont les états sont le nœuds choisis, et la probabilité de transition est la probabilité de passer à d'un nœud à

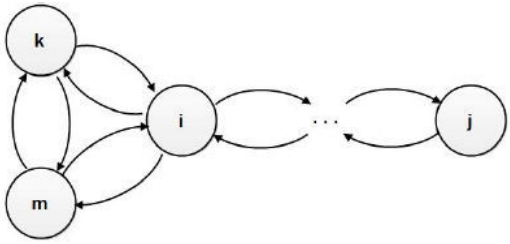


Figure 3: Exemple de chaîne de Markov représentant
le processus d'échantillonnage

l'autre. Ainsi la probabilité de choisir un nœud est équivalent à la probabilité d'état de la chaîne de Markov. Quand la chaîne de Markov est stationnaire, la distribution empirique d'état converge asymptotiquement, quand T est suffisamment grand, vers une distribution limite stationnaire.

Le lemme suivant lie les propriétés du graphe et de la chaîne de Markov représentant l'échantillonnage.

Lemme 1- *Si le graphe échantillonné contient au moins un nœud avec un coefficient de clustering non nul et une matrice de transition telle que* $Pr\{p_{i,j} > 0 | p_{j,i} > 0\} = 1$, *alors la chaîne de Markov est ergodique et irréductible.*

Ce lemme permet de prouver le théorème suivant qui donne les conditions nécessaires et suffisantes pour qu'un échantillonnage sans biais soit possible.

Théorème 1- *Si le graphe échantillonné contient au moins un nœud avec un coefficient de clustering non nul et une matrice de transition telle que* $Pr\{p_{i,j} > 0 | p_{j,i} > 0\} = 1$, *alors la condition nécessaire et suffisante pour avoir une méthode d'échantillonnage sans biais est la suivante :*

$$\forall i \in V, \sum_{j=1}^{|V|} P_{j,i} = 1$$

L'existence de nœuds dissasortifs rend difficile la conception d'algorithme d'échantillonnage. Ces nœuds ont un degré important et sont encerclés de beaucoup de nœuds de degré faible. Dans ce contexte deux

approches sont applicables afin de valider la condition d'absence de biais définie dans le théorème 1. Une première approche, qui est à la base de l'intuition de MHRW consiste à laisser l'échantillonneur dans les nœuds de faible de degré [32]. La seconde solution consiste à choisir un ensemble de nœuds qui ne sont pas des voisins des diassortatifs et de permettre à l'échantillonneur de sauter vers ces nœuds [79][78]. Cette approche aboutit à un graphe échantillonné contenant de nombreux composants non-connexes plutôt qu'un sous-graphe bien connecté.

L'algorithme de marche aléatoire de Metropolis-Hastings (MHRW) utilise une probabilité pour l'échantillonneur d'aller d'un nœud u à son voisin v qui est égale à :

$$P_{u,v} = \begin{cases} \min\left(\frac{1}{k_u}, \frac{1}{k_v}\right), & \text{if } v \text{ is a neighbor of } u \\ 1 - \sum_{y \neq u} P_{u,y}, & \text{if } v = u \end{cases}$$

Il est facilement vérifiable que pour chaque deux nœuds voisins u et v $P_{u,v} = P_{v,u} = \min\left(\frac{1}{k_u}, \frac{1}{k_v}\right) > 0$. De plus on peut vérifier que $\forall i \in V, \sum_{j=1}^{|V|} P_{j,i} = \sum_{j=1}^{|V|} P_{j,i} = 1$. Ce qui montre que les conditions du théorème 1 sont remplies et que MHRW fourni bien un échantillonnage sans biais. MHRW réussi un échantillonnage sans biais au prix du choix répété de nœuds à bas degré.

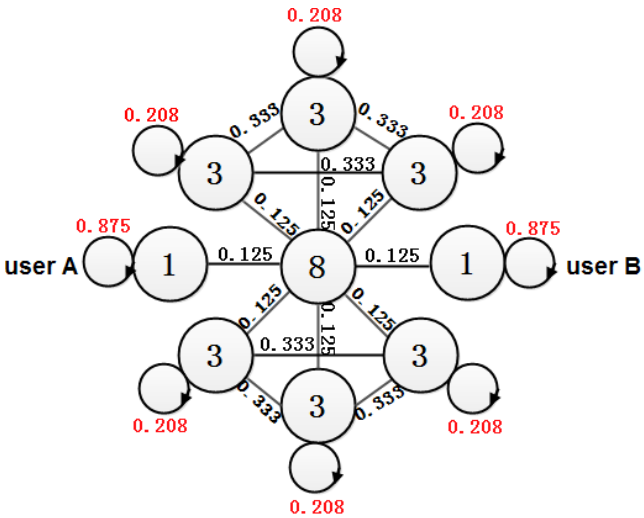


Fig. 2- Probabilité d'auto-échantillonnage dans MHRW

La probabilité d'aller d'un nœud i à lui même, $P_{i,i}$, est appelé probabilité d'auto-échantillonnage. Cette probabilité dans MHRW est élevée.

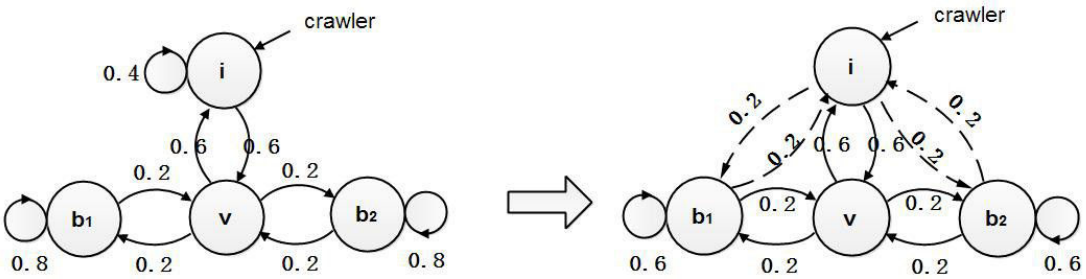


Fig 3-Example illustrant l'ajout de liens fictis.

2.2 Echantillonnage sans biais avec liens fictifs

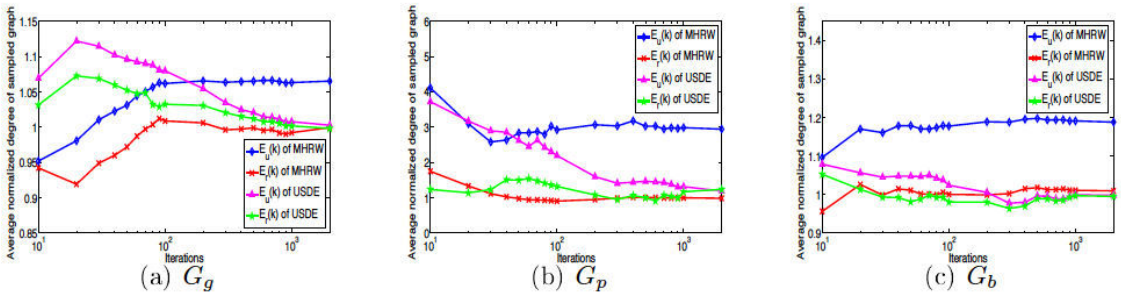
J'ai développé une nouvelle approche d'échantillonnage appelée USDE. Cette approche garde le caractère sans biais de MHRW tout en évitant l'auto-échantillonnage. USDE atteint cet objectif en ajoutant des liens fictifs au graphe. Il est important de considérer que ces liens ne sont pas réellement ajoutés au graphe. Ils ne sont considérés que pour l'échantillonnage.

La figure 3 représente un exemple d'ajout de liens fictifs. Afin que la contrainte d'échantillonnage sans biais reste valable il convient de s'assurer que le graphe après l'ajout de liens fictifs valide encore les contraintes définies par le théorème 1. Ceci est atteint en redistribuant la probabilité d'auto-échantillonnage d'un nœud sur les nœuds fictifs. Le seul problème est qu'on ne connaît pas *a priori* la probabilité d'auto-échantillonnage d'un nœud avant de l'avoir visité. USDE utilise plutôt que cette probabilité une estimation basse de celle ci calculée en utilisant seulement les informations des voisins. Cette borne basse est égale à :

$$LP_i = \sum_{v \in \{S(i) \cap V'\}} \left(\frac{1}{k_i} - \frac{1}{k_v}\right), \quad \text{where } k_v > k_i$$

où $S(i)$ est l'ensemble des voisins et V' est l'ensemble des nœuds visités.

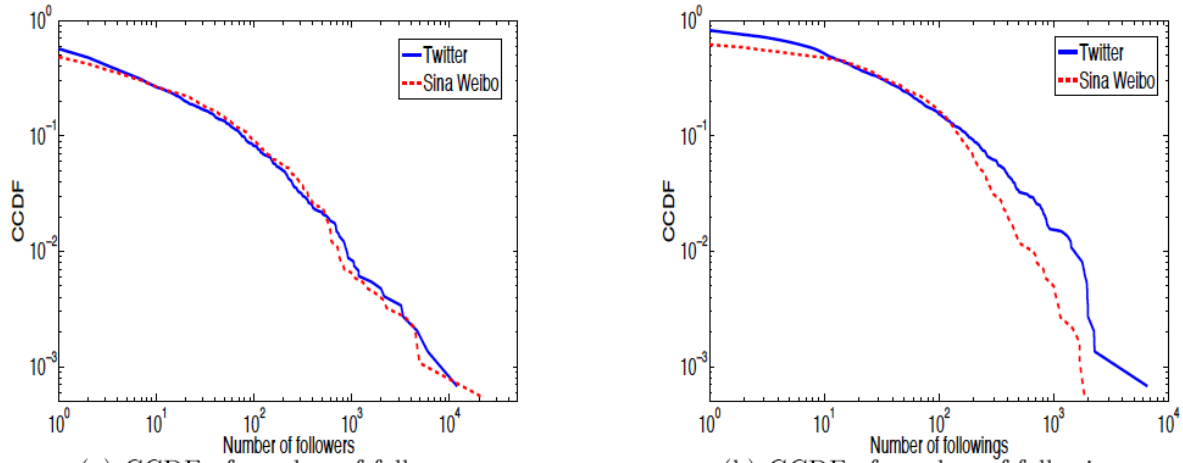
La performance d'un échantillonnage suivant USDE et celle de MHRW sont comparées sur 3 graphes aléatoires, doit l'un est un graphe de Barbell.



(a) G_g (b) G_p (c) G_b

Fig.4 Ratio du degré moyen obtenu sur le graphe échantillonné par différente méthodes d'échantillonnage

(Gb). La figure 4 compare le degré moyen estimé sur les graphes échantillonnés par différentes méthodes en représentant le ratio entre le degré estimé sur graphe échantillonné et le degré moyen réel mesuré sur la totalité du graphe. L'estimation se fait de deux façons. Une première méthode utilise les répétitions d'un même nœud dans l'estimation ($E_r(k)$) et une seconde n'utilise pas les répétitions ($E_u(k)$). On peut observer que pour MHRW la valeur $E_u(k)$ ne converge pas vers la vrai valeur mais que pour USDE cette convergence a bien lieu. De plus cette convergence est de 1.5 à 3 fois plus rapide pour USDE.



(a) CCDF of number of followers (b) CCDF of number of followings

Fig.6- Distribution Cumulative Complémentaire du nombre de suivant et de suivi estimé sur le graphe échantillonné

3- Modélisation de la diffusion d'information dans les microblogs.

1- Le modèle de cascade multiplicative pour les tweets

Un utilisateur de micro-blog peut suivre un autre utilisateur, *i.e.*, il recevra ainsi tous les tweets émis par cet utilisateur. Les mécanismes de microblog sont fondés sur le mécanisme de retweet, *i.e.*, un utilisateur choisi de renvoyer à toutes les personnes le suivant un message qu'il a reçu. Ce mécanisme permet de diffuser l'information dans les microblogs.

Le processus de retweeting peut être décrit par une cascade multiplicative aléatoire. Formellement, un processus de cascade multiplicative $X(\ .\ )$ peut être décrit en chaque point k comme une multiplication de n variables aléatoires indépendantes et identiquement distribués $m_1, \dots, m_n$, *i.e.*, $X(k) = m_1 \times m_2 \times \dots \times m_n$. Supposez que le nombre de nouveaux utilisateurs qui retweetent un message à l'étape i est un coefficient α_i du nombre total de personnes qui ont retweeté le message jusqu'à l'étape (i-1). Ainsi le nombre total d'utilisateur qui retweetent un message est après n étapes, $N^n(t)$, est obtenu par :

$$N^n(t) = (1 + \alpha_1) \times (1 + \alpha_2) \times \dots \times (1 + \alpha_n)$$

Ce qui est une cascade multiplicative. Le coefficient α_i dépend de deux paramètres : la proportion de suivant qui vont retweeter le message à l'étape i et le degré sortant, *i.e.*, le nombre de suivant, des personnes retweetant.

Similairement au théorème de limite centrale pour une somme de variable aléatoire, la distribution asymptotique d'une cascade multiplicative peut être obtenue et suit une distribution exponentielle étendue (*Stretched Exponential*) [30] :

$$P\{X \geq x\} = e^{-\left(\frac{x}{x_0}\right)^c}$$

où le paramètre c est lié au nombre de cascades, *i.e.*, nombre de variables multipliés, m, par la relation suivante $c = \frac{1}{m}$ et x_0 est un paramètre lié à l'échelle

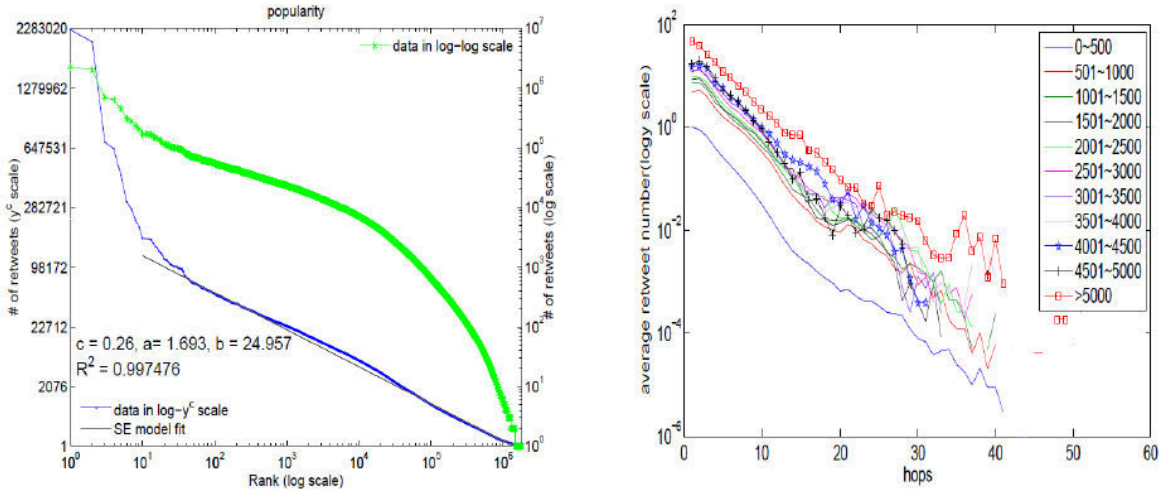


Fig. 7- Calibration d'un modèle exponentiel élargi

du processus [56]. La distribution précédente a une forme qui ressemble fortement à une loi de puissance, et pour cette raison elle est fréquemment prise par erreur pour cette dernière. Mais il existe un moyen de détecter un loi exponentielle élargie. Soit i le rang d'une observation venant d'une distribution exponentielle élargie et y_i sa valeur. Si les données sont issues d'un distribution exponentielle élargie alors la relation suivante sera validée :

$$y_i^c = -(x_0^c)\log i + y_1^c$$

Ainsi, si des observations empiriques suivent la relation log-linéaire précédente ont peu raisonnablement considérer qu'elles sont issues d'une distribution exponentielle élargie. Je présente dans la Figure 7 la calibration du nombre de retweet observé empiriquement dans le réseau de microblogging Weibo Sina. La Figure atteste de la qualité de la calibration qui est évalué par le biais du R^2 =0.997. Seul les points initiaux de la courbe de suivent pas la distribution attendue. Ceci peut être expliqué par l'effet du roi [56], *i.e.*, les personnes très populaires sont largement plus populaires qu'attendu.

2- Modèle de Galton-Watson avec extinction

Le modèle de cascade multiplicative permet de bien décrire le processus de retweet, mais il ne donne pas un modèle explicatif permettant d'intégrer des paramètres expliquant la diffusion. Dans la suite j'ai développé un modèle explicatif plus fin pour ce processus qui prend en compte la topologie du

réseau de diffusion ainsi que les caractéristiques des contenus diffusés. Ce modèle s'inspire du modèle de branchement de Galton-Watson (GW) qui a été utilisé pour étudier l'évolution des noms de familles célèbres [76]. Le processus GW est un processus de branchement $\{X_n\}$ où X_n représente le nombre d'utilisateurs recevant un tweet particulier par un chemin de n hops de retweet. Le processus $\{X_n\}$ évolue suivant la formyle de récurrence suivante.

$$X_0 = 1, X_{n+1} = \sum_{j=1}^{X_n} \xi_j$$

où pour chaque génération n, ξ_j est une séquence de variables aléatoires indépendantes et identiquement distribuées de distribution f (k), lié à la distribution du nombre de suivant d'une personne dans le réseaux de microblog. Dans ce cas nous avons :

$$f(k) = (1-\alpha)\mathbb{I}_{\{k=0\}} + \alpha\mathcal{D}(k)\mathbb{I}_{\{k>0\}}$$

où α est la probabilité qu'un utilisateur recevant un tweet le retweet, et $\mathcal{D}(k)$ est la distribution de degré de du réseau social du microblog, *i.e.*, la distribution du nombre de suivant d'une personne dans le réseaux de microblog.

Néanmoins le processus de retweet a une différence principale avec celui de Galton-Watson. Alors que le processus GW peut être infini, le processus de retweet est fini, et l'intérêt de retweeter un message décroit avec le temps et la diffusion s'arrête. Il convient donc d'ajouter au modèle une probabilité d'extinction π qui représente la probabilité que le processus GW se termine prématurément à la génération n. Ceci aboutit à la définition d'un processus de Galton-Watson avec extinction (*Galton-Watson with Killing, GWK*).

Un processus GWK peut être étudié par les méthodes classiques d'analyse des arbres de GW. En particulier la fonction génératrice de probabilité de X_n peut être calculé récursivement par la relation suivante : $\phi_{n+1}(s) = \phi_n(\phi(s))$, où $\phi(s)$ est la fonction génératrice de probabilité du nombre de suivant dans un réseau de diffusion. Quand une probabilité d'extinction est ajouté le la fonction génératrice devient :

$$\phi_M(s) = \sum_{n=1}^{\infty} \phi_m(s)\,\pi(1-\pi)^n$$

Le nombre moyen de personne recevant le tweet ($\overline{M}$) peut être estimé grâce à la relation suivante :

$$\overline{M} = \frac{\mu\pi}{1-\mu+\mu\pi}$$

si $\mu_K = \alpha\delta < 1$, où α est la probabilité de retweet et δ le nombre moyen de suivant des personnes ayant retweeté le message. Une analyse asymptotique de la queue de la distribution du nombre de personnes ayant reçu le message prédit que cette queue suivra une loi de puissance avec un exposant égal à :

$$1 - \frac{\log(1-\pi)}{\log\mu}$$

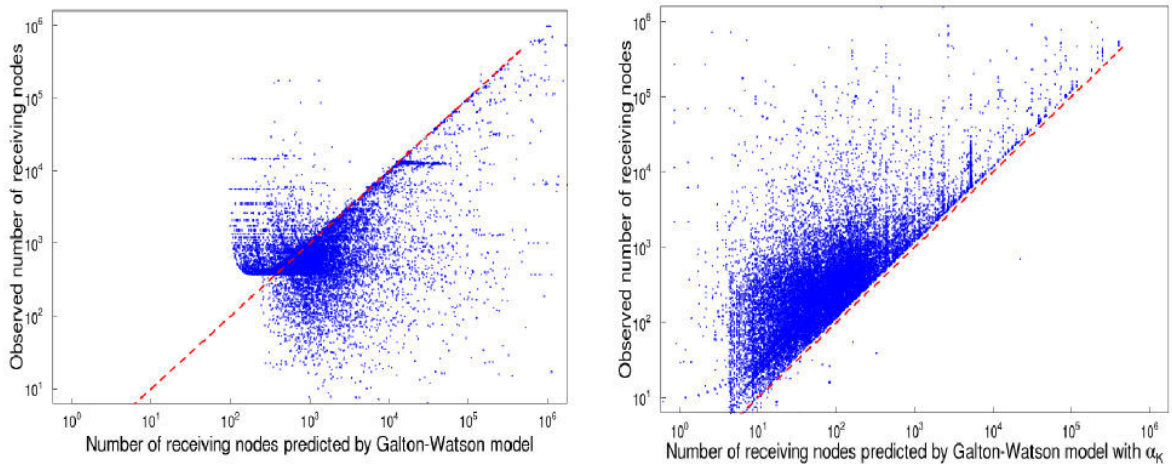


Figure 8- le nombre d'utilisateurs recevant un message par rapport au nombre d'utilisateurs prédit par un modèle de GW (à gauche) et un modèle GWK (à droite).

Le modèle GWK peut être évalué. Je présente dans la Figure 8 le nombre d'utilisateurs recevant un message par rapport au nombre d'utilisateurs prédit par un modèle de GW (à gauche) et un modèle GWT (à droite). La figure montre bien l'impact du paramètre d'extinction sur la qualité de l'estimation.

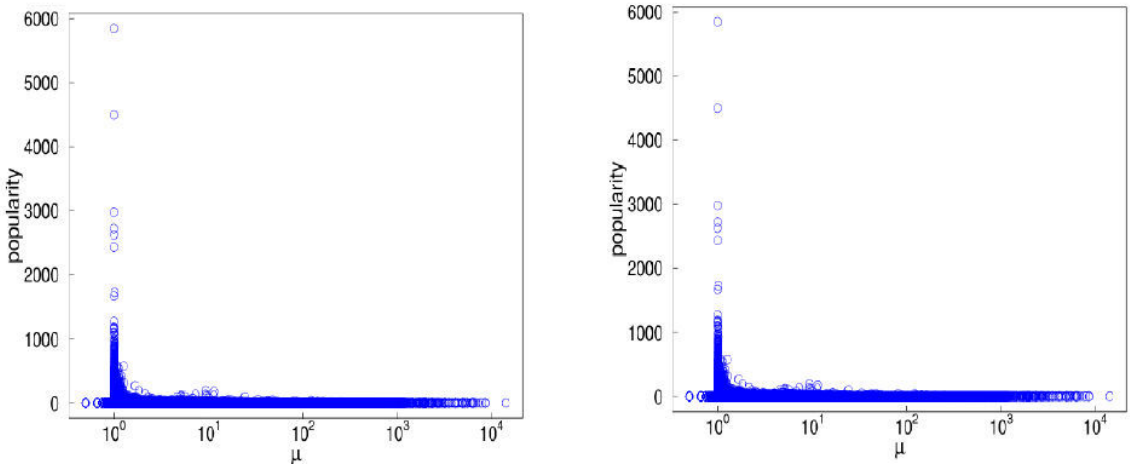


Fig. 9 Popularité d'un tweet en fonction de μ mesuré sur Twitter et sur Weibo

Le modèle peut être utilisé pour prédire la popularité d'un tweet. Je présente dans la figure 9 la relation entre le paramètre μ et la popularité mesuré en terme de nombre de personnes recevant le message. La figure montre que tous les tweets populaire sont concentrés autour des valeurs de μ=1. Ce qui montre que les ingrédients d'une large diffusion d'un tweet se trouvent à la croisée d'une probabilité de retweet importante, *i.e.*, d'un intérêt intrinsèque du tweet, et d'un bon réseau de diffusion avec des utilisateurs ayant un nombre moyen de suivant important mais pas trop grand (ce qui aboutirait à un μ largement supérieur à 1).

The Analysis and Applications of Information Diffusion in Microblogs

by

Dong Wang

Submitted to the Université de Grenoble
on October 20, 2015, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Microblog service (such as Twitter and Sina Weibo) has become an important platform for Internet content sharing, leading to a new way of online information diffusion which is different from the traditional Word-of-Mouth spreading. As the information in Microblog is used widely in public opinion mining, viral marketing and political campaigns, understanding mechanisms describing how information diffuses over Microblogs, and explaining how some tweets become popular, are meaningful in order to analyze the evolution of this new social medium in the future.

The analysis of the information diffusion model in Microblog involves the data collection from Microblog, the modeling on information spreading and the applications of the model. At first, it is hard for researchers to get and deal with the complete dataset of Microblog, given the huge amount of data. Therefore, how to design an efficient and unbiased sampling algorithm for Microblog is essential. Besides, the retweeting process in Microblog is complicated which is relative to the ephemerality of information quality, the topology of Microblog network and the features of publisher and retweeters (such as number of followers). As a result, the two traditional models of information diffusion, Independent Cascades model and Linear Threshold model, can not describe the retweeting process in Microblog accurately. Given this fact, the analysis and design of a new model to characterize the information diffusion in Microblog is necessary. Finally, the comprehensive description of the correlation between the information diffusion in Microblog and the searching trends of keywords on search engines is lacked although some work has been found this relationship preliminarily. The application prospect on web of the information diffusion model is still unclear.

This work makes a complete analysis of information diffusion in Microblog from these three aspects accordingly. To sum up, the contributions and innovations of the work are as follows:

1)The two popular unbiased Online Social Network (OSN) sampling algorithms, Metropolis-Hastings Random Walk (MHRW) and Unbiased Sampling method for Directed Social Graph (USDSG), are likely to yield considerable self-sampling prob-

abilities when they are used in Microblog where the local disassortativity is obvious, suffering from inefficiency and low quality of samples. To solve this problem, this work models the process of OSN sampling as a Markov process and deduces the necessary and sufficient condition of unbiased sampling. Based on this unbiased condition, the work proposes an effective and unbiased sampling algorithms, Unbiased Sampling method with Dummy Edges (USDE), which reduces the self-sampling probabilities of MHRW and USDSG by amortizing such probabilities to the moving probabilities between different nodes uniformly while keeping the unbiased condition. The results of experiments demonstrate that the average node degree of samples of MHRW and USDSG is 2 - 4 times as high as the ground truth while USDE can provide the approximation of ground truth when the sampling repetitions are removed. In the aspect of sampling efficiency, the average sampling time per node in USDE is only a half of the one in MHRW and USDSG.

2)This work targets at the shortages of Independent Cascades (IC) model and Linear Threshold (LT) model in characterizing the retweeting process in Microblog and introduces a Galton Watson with Killing (GWK) model which considers all the three important factors including the ephemerality of information quality, the topology of network and the features of publisher and retweeters accurately. The work validates the applicability of GWK model over two datasets from Sina Weibo and Twitter and the results show that GWK model can fit 82% numbers of receivers of information and 90% maximum numbers of hops in the real retweeting process. Besides, the GWK model is useful for revealing the endogenous and exogenous factors which affect the popularity of tweets.

3)The work makes a comprehensive analysis of the correlation between the popularity and trendiness of topics in Microblog and the search trends on search engines. The results show that individual topics in Twitter and in the web share similar trending patterns both from the temporal and the spatial aspects. Nevertheless, the trendiness in Twitter can precede for a few hours and is highly unstable compared to the one in web. These features indicate the likelihood that the topics in Microblog can be used as superior adwords in Search Engine Marketing (SEM).

4)Motivated by the correlation between the popularity and trendiness of topics in Microblog and the search trends on search engines, the work makes an economic analysis of the market involving a third-party ad broker, which is a popular market in current SEM, and finds that the adwords augmenting strategy with the trending and popular topics in Twitter enables the broker to achieve, on average, four folds larger return on investment than with a non-augmented strategy, while still maintaining the same level of risk.

Thesis Supervisor: Kavé Salamatian, Mohamed-Ali Kaafar and Gaogang Xie
Title: Professor

Acknowledgments

First,I would like to show my deepest thank to my three supervisors, Dr. Kavé Salamatian, Dr. Mohamed-Ali Kaafar, and Dr. Gaogang Xie, who have provided me with useful guidance in every stage of my Ph.D researches. Without their enlightening instruction,impressive kindness and patience,I could not have completed my thesis. Besides, they have also gave me valuable advices on my English skill, my life and my career search, which will benefit me for the whole life.

I shall extend my warmest thanks to Dr. Zhenyu Li in Institute of Computing Technology, Chinese Academy of Sciences, who has helped me to develop the fundamental and essential academic competence and provided me with the superior research environment.

Finally, I'd like to express my special thank to my parents, for their deep love, encouragement and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation and Background

As the development of technologies on WEB 2.0, Microblog service (such as Twitter and Sina Weibo) has been becoming a crucial platform for online information sharing. In Microblog, users can follow any others and post their messages, called tweets, the size of which is limited (to 140 characters in case of Sina Weibo and Twitter). Followers in Microblog services can retweet some of the tweets received from their followings and these retweets can be seen by their own followers. The features of Microblog services lead to a new pattern of online information diffusion which is quite different with the traditional word-of-mouth spreading and have a great influence on the pattern of web access and Internet resources discovery.

At first, the retweeting mechanism of Microblog accelerates the information propagation recently. Figure 1-1 shows the relative traffic in Twitter in one hour after the "East Japan Earthquake" where the colorful lines represent the relative tweets published from Japan and the white ones are the replies of these tweets[98]. In only one hour, there are more than 5,000 relative tweets generated in Twitter every minutes.

Besides, the correlation between the trends of information diffusion in Microblog (which is the representative service for information sharing in WEB2.0) and search engines (which is the traditional platform for information acquisition in WEB1.0) becomes more and more significant. Figure 1-2 depicts the popularity trends of topic

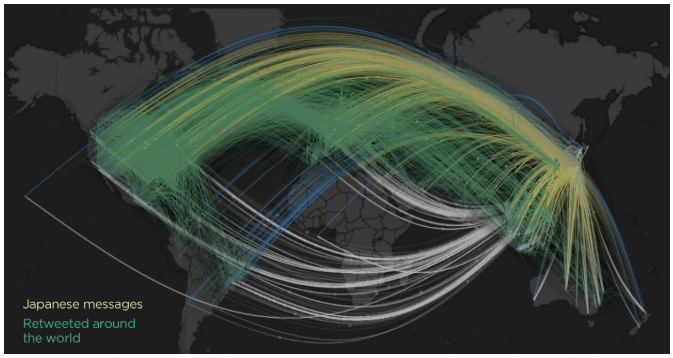


Figure 1-1: The relative traffic in Twitter in one hour after "Japan Earthquake"[98]

"Japan Earthquake" in Google and in Twitter in March,2011 respectively[40][95]. It is easy to find that this topic in Twitter and in Google shares similar trending patterns in the same period.

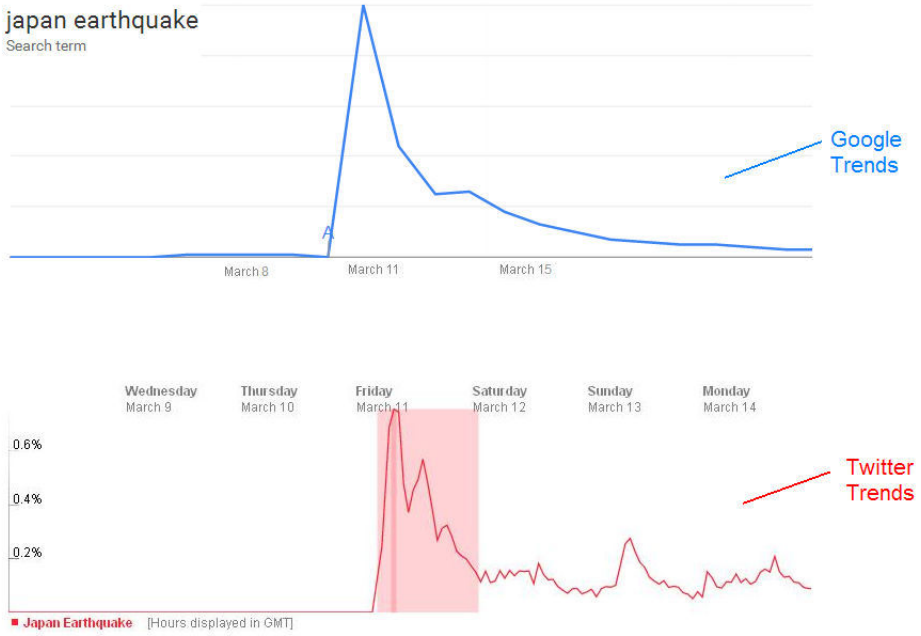


Figure 1-2: The trends of "Japan Earthquake" in Google and in Twitter in March, 2011[40][95]

These features make the analysis of information diffusion in Microblog meaningful in public opinion mining, online recommendation and the optimization of Search Engine Marketing (SEM), therefore, the study of information propagation in Microblog

has been becoming the attractive field among the OSN researches recently.

1.2 Research objectives

There are three steps to analyze the information diffusion in Microblog. At first, researchers should collect a representative and unbiased dataset from the Microblog services. And then, based on the collected dataset, the researchers observe the characters of the information diffusion in Microblog and find an appropriate mathematical model to describe the process of information spreading. Finally, to find the application of the information diffusion model in the evolution of Internet is necessary. However, there are lots of challenges in each of the three steps:

1)Sampling the data: Microblog services have become more and more large-scaled. There are over 200 million active users in Twitter as of Feb. 2013 [107] and the number of users in Sina Weibo, the largest Chinese Microblog service, has reached 300 million up to May. 2012 [86]. While using complete datasets provided by the official companies results the best results, it is hard for researchers to get such datasets as most companies are reluctant to share their data in order to protect users' privacy. Besides, it may require unreasonable time to calculate the results, given the huge amount of data. Thus, an effective and accurate sampling method is necessary.

2)Analyzing and modeling: Unlike the traditional word-of-mouth pattern, the information diffusion in Microblog is a complicated process which is related to the quality of topics, topology of network and the social features of users. However, the pervious models of information diffusion such as IC model and LT model can't describe this process exactly and a new appropriate model is essential for researches.

3)Applying the model: The social nature of the Web 2.0 leads to new patterns of web access and Internet resources discovery. Although there are several studies addressed the interaction between online social media and the popularity of online digital content, the relationship that might exist between online information diffusion in Microblogs and web interest has been overlooked, *i.e.* how to apply the model of online information diffusion to the traditional web platform (specially, search engines)

should be explored carefully.

1.3 Contributions

This work makes a comprehensive analysis of information diffusion in Microblog from these three aspects accordingly. To sum up, the contributions and innovations of the work are as follows:

1)The two popular unbiased OSN sampling algorithms, MHRW and USDSG, are likely to yield considerable self-sampling probabilities when they are used in Microblog where the local disassortativity is obvious, suffering from inefficiency and low quality of samples. To solve this problem, this work models the process of OSN sampling as a Markov process and deduces the necessary and sufficient condition of unbiased sampling. Based on this unbiased condition, the work proposes an effective and unbiased sampling algorithms USDE, which reduces the self-sampling probabilities of MHRW and USDSG by amortizing such probabilities to the moving probabilities between different nodes uniformly while keeping the unbiased condition. The results of experiments demonstrate that the average node degree of samples generated by MHRW and USDSG is 2 - 4 times as high as the ground truth while USDE can provide the approximation of ground truth when the sampling repetitions are removed. In the aspect of sampling efficiency, the average sampling time per node in USDE is only a half of the one in MHRW and USDSG.

2)This work targets at the shortages of IC model and LT model in characterizing the retweeting process in Microblog and introduces a GWK model which considers all the three important factors including the ephemerality of information quality, the topology of network and the features of publisher and retweeters accurately. The work validates the applicability of GWK model over two datasets from Sina Weibo and Twitter and the results show that GWK model can fit 82% numbers of receivers of information and 90% maximum numbers of hops in the real retweeting process. Besides, the GWK model is useful for revealing the endogenous and exogenous factors which affect the popularity of tweets.

3)The work analyzes the correlation between the popularity and trendiness of topics in Microblog and the search trends on search engines. The results show that individual topics in Twitter and in the web share similar trending patterns both from the temporal and the spatial aspects. Nevertheless, the trendiness in Twitter can precede for a few hours and is highly unstable compared to the one in web. These features indicate the likelihood that the topics in Microblog can be used as superior adwords in Search Engine Marketing (SEM).

4)Motivated by the correlation between the popularity and trendiness of topics in Microblog and the search trends on search engines, the work makes an economic analysis of the market involving a third-party ad broker, which is a popular market in current SEM, and finds that the adwords-augmented strategy with the trending and popular topics in Twitter enables the broker to achieve, on average, four folds larger return on investment than with a non-augmented strategy, while still maintaining the same level of risk.

1.4 Structure

The structure of this paper is as follows: chapter 2 derives the sufficient and necessary condition of unbiased sampling and proposes an effective and unbiased sampling method based on the unbiased condition to collect the Microblog dataset. Chapter 3 analyzes the features of information diffusion in Microblog, and introduces a Galton-Watson-based explicative model to describe the process of information spreading. And then chapter 4 checks the correlation between the information diffusion in Microblog and web interests, and based on the correlation discovered in chapter 4, the potential of information in Microblog in optimization of SEM is explored. Finally, the work is concluded in chapter 6.

Chapter 2

Unbiased Sampling Analysis

2.1 Motivation

Social media systems have gained tremendous popularity in the past few years. For example, Twitter, the world's largest Microblog service, has around 200 million active users by Feb. 2013 [107]. Sina Weibo, the largest Chinese Microblog website, has more than 300 million registered users as of May. 2012 [86].

Users in social media systems are allowed to follow or subscribe to others. Therefore, users in a system form a network, where vertices are users and edges represent the following or subscription relationships. Given the importance of content sharing and information diffusion in social media systems [18][6][69][58], there have been great interests in analyzing these networks. However, the huge size of such networks makes it very hard to get a snapshot of the complete network and the alternative is to obtain representative or "unbiased" samples.

Generally, there are two measures of "unbiased" for a sample. One emphasizes on unbiased sampling on edges [79][77], *i.e.* individual edges are sampled with uniform probability. Such samples could provide unbiased estimations of global characteristics on edges (*e.g.* global clustering coefficient). The other one on the other hand emphasizes on unbiased sampling on nodes [34][93][104], *i.e.* individual nodes can be sampled with uniform probability. Since online social media systems are more user-centric, this work considers the second measure, *i.e.* unbiased sampling on nodes.

A representative sample can be used either to estimate the topological characteristics (*e.g.* node degree), or to provide unbiased and well-connected subgraphs as a basis for the long-term information propagation and user behavior analysis, or even both. Indeed, a real unbiased and well-connected subgraph is not necessary if the focus is only on the estimation of topological characteristics [81][80]. However, it is mandatory for the studies on long-term user behavior analysis (*e.g.* retweeting tweets, commenting *etc.*) and information propagation models [87][92], as getting an unbiased and well-connected graph is the first step towards these goals. In addition, unbiased and well-connected subgraphs are also important for re-analysis or re-validation in the future [35][74]. As such, the work studies sampling methods that can provide unbiased and well-connected samples, *e.g.* Metropolis-Hastings Random Walk (MHRW) [34] and its variations [35][104][74].

MHRW and its variations are tailored for online social networks (*e.g.* Facebook), which are *friendship-based* networks (*i.e.* users are likely to connect with those having similar number of friends). The intuition behind is to avoid sampling high-degree nodes by explicitly increasing the probability of sampling low-degree nodes during sampling process. They suffer from two limitations when applied in sampling online social media (*e.g.* Twitter), which are *content-driven* networks (*i.e.* a user follows another mainly because the other side provides content of his interests). First, the content-driven nature of online social media leads to the fact that users tend to follow power-users with degrees orders of magnitude larger [69][58]. As such, high-degree nodes tend to be surrounded by plenty of low-degree nodes, and thus show local disassortative mixing pattern. According to MHRW and its alternatives, sampling crawlers will be trapped in the low-degree nodes for a long time and unable to find new nodes in such networks quickly. The second limitation is related to the high sampling repetitions on low-degree nodes. While the repetitions could be counted several times for topological characteristics estimations, in the context of individual user behavior studies, a sampled node can only be counted once no matter how many times it was sampled. It has been found in [81][35][74] that MHRW fails in providing unbiased samples if repetitions are removed.

The above facts motivate researchers to study unbiased sampling methods for online social media. This work aims to provide good quality of samples (measured by the ability to provide unbiased and well-connected subgraphs) with high sampling efficiency (measured by the convergence of crawlers and the speed in discovering new nodes and new user attributes). In detail, the work first studies the sufficient and necessary condition for unbiased sampling of large-scale networks, and then analyzes the limitations of MHRW. Besides, the work proposes a novel unbiased sampling method and applies it to three synthetic networks and two real-life social media. To sum up, this work makes the following contributions:

- The work models the process of random walk-based online sampling as a Markov chain and concludes the sufficient and necessary condition for unbiased sampling. The condition could be used as a guideline for the design of various unbiased sampling methods. The work further analyzes the performance issues of MHRW when sampling online social media because of the high local disassortativity.
- The work proposes a novel unbiased sampling method, called Unbiased Sampling method with Dummy Edges (USDE), for online social media. The method explicitly adds dummy edges between low-degree nodes that have high self-sampling probabilities in MHRW. The dummy edges enable flexible moves of sampling crawlers from low-degree nodes to unvisited ones while keeping the connectivity of samples. Therefore, the addition of dummy edges improves the sampling performance compared with MHRW.
- The work conducts extensive sampling experiments in three synthetic networks to evaluate the performance of USDE in terms of quality of samples and sampling efficiency. The results show that USDE can provide more unbiased samples than MHRW, and keep the connectivity of samples, outperforming the unbiased sampling methods with random jumps proposed in [80][8]. In terms of sampling efficiency, USDE reduces the average sampling times per node by 50% compared with MHRW and is 1.5 times as efficient as MHRW in terms of discovering new user attributes.

- The work applies the proposed method to sample two popular social media systems: Twitter and Sina Weibo. The results further demonstrate that USDE performs well from the perspectives of both quality of samples and sampling efficiency. For example, with 1,000 sampling iterations in Twitter, the average sampling times per node in USDE is only one third of that in MHRW. Besides, USDE could identify more than 200 geolocation categories with 1,000 iterations while this number is only 50 for MHRW.

2.2 Related work

Breadth First Search (BFS) is one of the popular sampling methods that has been used in [69][4][101][20] to collect datasets of online social networks. Random walk, another kind of sampling methods, has also been widely used in sampling unbalanced heterogeneous bipartite graphs[109], directed graphs [66][80], and recently in sampling online social networks[54][52].

BFS and random walk are shown to be biased towards high-degree nodes and the statistical properties obtained from these samples directly are inaccurate[61][9][112][55][65]. To address this problem, several sampling methods have been proposed to generate unbiased samples. Stutznach *et al.*[93] used Metropolized Random Walk with Backtracking (MRWB) to select representative samples of Peer-to-Peer networks. Following it, Gjoka *et al.*[34] proposed the Metropolis-Hastings Random Walk (MHRW) to get unbiased samples of Facebook. Wang *et al.*[104] found that by taking the unidirectional edges as bidirectional edges, MHRW is also suitable for sampling directed graphs.

Recent studies [81][35][74][102] have pointed out that MHRW suffers from inefficiency in discovering new nodes and might not provide accurate topological characteristics when the sampled nodes are considered uniquely (*i.e.* sampling repetitions are removed). This weakness is especially significant in the networks where the local disassortativity is obvious, which is a major characteristic of online social media.

Besides unbiased sampling from the perspective of nodes, Rasti *et al.*[77] and

Ribeiro *et al.*[81] showed that the unbiased sampling on edges can estimate the global characteristics on edges (*e.g.* global clustering coefficient) accurately. Ribeiro *et al.* [79] introduced multidimensional random walks for the unbiased sampling on edges of directed graphs.

Some other researches focus on the estimation of characteristics of the complete network based on the samples obtained by the biased sampling methods such as BFS and Random Walk. Kurant *et al.* [56] analyzed the principle of bias in BFS and random walk-based sampling methods and proposed unbiased estimators to compute the distribution of degree. Dasgupta *et al.* [30] studied the estimation of the average node degree in social graphs based on random walk using Hoeffding inequality and Bernstein inequality. Ribeiro *et al.* [81] showed that the topological characteristics of directed graphs can also be estimated from the samples obtained using random walk. They proposed to use random jumps in random walk-based sampling to reduce the estimation error for both undirected graphs [8] and directed graphs [80]. While estimation of node degree is the most important aspect for researches only interested in topological characteristics, for those interested in long-term user behavior and information propagation analysis in social systems [87][92], unbiased and well-connected subgraphs are mandatory as such subgraphs are the basis for further studies.

Another type of sampling method is Uniform Sample technique (UNI), which is always used to obtain ground truth of node-related properties [45]. For example, if the distribution of individual nodes' identifiers (IDs) and the ID space are known in advance, one can get a uniform sample of IDs and then obtain a uniform sample of the network by accessing the degree or user attributes of each selected ID. UNI is inefficient in online social media because the user ID space is always sparse [34] and the samples obtained are weak-connected.

2.3 On Random Walk-based Unbiased Sampling of Online Social Media

The content-driven nature of online social media leads to a low level of reciprocity [58]. For example, users might not follow their followers back in Twitter. These networks are therefore abstracted as directed graphs. From the perspective of sampling, it has been shown that the random walk with "backward edge traversals" which allows the crawler to consider the unidirectional edges as bidirectional ones in directed graphs can achieve the similar performance to the one in the corresponding undirected graphs [81][79][80][104]. This work uses the similar idea to study the sampling of online social media. Taking Twitter as an example, the work treats the unidirectional edges representing follower and following relationships as bidirectional ones. Then the Twitter network is viewed as an undirected graph $G = (V, E)$, where V is the set of nodes representing users, E is the set of bidirectional edges. In this graph, the degree of node i, k_i, is the number of neighbors connected with i via either following-ship or follower-ship.

A prominent feature of online social media is that in the abstracted graphs, nodes with degree orders of magnitude large might be surrounded by plenty of low-degree nodes, implying a high local disassortativity. This feature impairs the performance of previously proposed sampling methods (*e.g.* MHRW) tailored for online social networks (*e.g.* Facebook), where the local assortative mixing pattern is dominant. This section first introduces the quantitative measure of local mixing pattern and then proceeds to the mathematical model of random walk-based sampling process which could help researchers design unbiased sampling methods. Finally, the work analyzes the performance problem of MHRW when the local disassortativity is high.

2.3.1 Background: Local Mixing Pattern

Mixing pattern, which can be classified broadly as assortative or disassortative, is a characteristic of network, referring to the extent for nodes to connect to other similar

or different nodes. Though the specific measure of similarity may vary, node degree is often used. If nodes tend to be connected with other nodes with similar degrees, assortative mixing pattern exists. Otherwise, there is a disassortative mixing pattern.

To capture the local tendency of connections for individual nodes, Piraveenan *et al.* [75] define the local assortativity metric. For a node of degree $(j+1)$, its local assortativity coefficient ρ is defined as follows:

$$\rho = \frac{j(j+1)(\bar{k} - \mu_q)}{2M\sigma_q^2} \tag{2.1}$$

where $\bar{k}$ is the average remaining degree of the node's neighbors, M is the number of links in the network, μ_q and σ_q are the mean and standard deviation of the remaining degree distribution of the network respectively. A positive (*resp.* negative) coefficient ρ indicates an effect of local assortativity (*resp.* disassortativity). When a high-degree node i is surrounded by a large number of low-degree nodes, the value of average remaining degree of the i's neighbors, $\bar{k}$, is likely to be less than the global average of remaining degree distribution μ_q. In this case, ρ of the high-degree node i is negative but with a large absolute value, showing a high local disassortativity.

The global assortativity coefficient for a graph $r = \sum_{i=1}^{|V|} \rho_i$, where $|V|$ is the number of nodes. As shown in [75], there might exist a large number of local disassortative nodes in a network, regardless of whether the network is overall assortative ($r > 0$), non-assortative ($r = 0$), or disassortative ($r < 0$).

2.3.2 Modeling the Random Walk-based Sampling Process

Here the work focuses on the random walk-based sampling methods, in which the crawler starts from a seed node and follows the edges to move towards others with predefined strategies. At each step, all neighbors of the current sampled node are potential candidates of the next step. This process carries on iteratively until enough nodes are sampled. The work aims at unbiased and well-connected samples in terms of nodes, rather than edges. In other words, a sampling algorithm can be considered as an unbiased one if the visiting probability of each node is uniform during the sampling

process and the sampled subgraphs are well-connected, as opposed to plenty of small connected components.

In random walk-based sampling methods, the probability that a node can be sampled at the step $T+1$ only depends on the sampled node at step T but not on the sampled nodes before T, which means the sampling is memorylessness. As such, the sampling process can be modeled as a Markov chain by taking the sampled nodes as states and the probability of moving from node i to j as the transition probability from state i to j. The probability that a node can be sampled at the step T is equivalent to the state distribution of the corresponding Markov chain at time T. If the Markov chain is stationary, the probability distribution will be convergent to the stationary distribution when T is large enough. Table 2.1 lists the mapping between the random walk-based sampling process and Markov chain. During the sampling process on a connected network, the crawler cannot stay at a node forever by re-sampling the same node. That said, $P_{i,i} < 1$, $\forall i \in V$.

Table 2.1: Mapping sampling process to Markov chain

Notation	Definition in sampling process	Definition in Markov chain
V	set of sampled nodes	state set
$P_{i,j}$	moving probability of the crawler from node i to node j directly	transition probability from state i to state j directly
$P_{i,j}^{(n)}$	moving probability of the crawler from node i to node j through n steps	transition probability from state i to state j through n states
π_i	probability that node i can be sampled	stationary distribution of state i
L_i	self-sampling probability of node i *i.e.* $P_{i,i}$	transition probability from state i to i directly *i.e.* $P_{i,i}$

It is noteworthy that the use of Markov chain modeling is only for proving the equivalence condition of unbiased sampling. The following Lemma can be proved for the Markov chain that models the sampling process on a graph having at least one node with non-zero clustering coefficient, where the clustering coefficient for a node is the ratio of the number of links that exist between its one-hop neighbors to the maximum number of links that could exist.

Lemma 1. *If the network for sampling contains at least one node with non-zero*

clustering coefficient and $P\{P_{j,i} > 0 | P_{i,j} > 0\} = 1$*, then the corresponding Markov chain is ergodic and irreducible.*

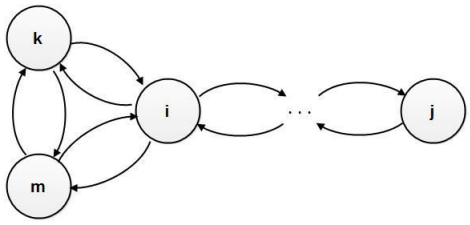


Figure 2-1: An example of Markov chain abstracted from sampling process

Proof. Supposing node i has a non-zero clustering coefficient as shown in Fig. 2-1, then, i can get back to itself by 2 transitions (through k or m) or 3 transitions (through k and m), *i.e.* both $P_{i,i}^{(2)}$ and $P_{i,i}^{(3)}$ are positive.

It is available to show that the node i can reach any state by exact $|V| - 1$ transitions as follows, where $|V|$ is the number of states. Clearly, i can reach any state through $|V| - 3$ transitions at most, *i.e.* i can reach any state through $|V| - 3 - 2n$ or $|V| - 3 - 2n - 1$ transitions, where n is a non-negative integer. Let $h_{i,j}$ denote the least number of transitions from i to j. If $h_{i,j} = |V| - 3 - 2n$, i can arrive at j through $|V| - 1$ transitions by adding $n+1$ loops of $i \to m \to i$, while if $h_{i,j} = |V| - 3 - 2n - 1$, i can arrive at j through $|V| - 1$ transitions by adding n loops of $i \to m \to i$ and 1 loop of $i \to m \to k \to i$.

Following this, it is easy to get that there is a positive integer $N = 2(|V| - 1)$ such that for each state in this Markov chain, it can reach any state by exact N transitions (using i as transfer point). That said, there exists $N \geq 1$ such that for any two states j, k, $P_{j,k}^{(N)} > 0$. This is the necessary and sufficient condition for a finite Markov chain to be ergodic and irreducible [73] . □

Based on the *Lemma 1*, the following necessary and sufficient condition for unbiased sampling based on random walk can be derived.

Theorem 1. *If the network for sampling has at least one node with non-zero clustering coefficient and* $P\{P_{j,i} > 0 | P_{i,j} > 0\} = 1$*, then the necessary and sufficient condition of being an unbiased sampling method is:* $\forall i \in V, \sum_{j=1}^{|V|} P_{j,i} = 1$.

Proof. Let's first prove the necessity. Supposing there is a stationary Markov chain, of which the State Transition Probabilities Matrix is as follows:

$$\begin{pmatrix} P_{1,1} & P_{1,2} & P_{1,3} & \dots \\ P_{2,1} & P_{2,2} & P_{2,3} & \dots \\ P_{3,1} & P_{3,2} & P_{3,3} & \dots \\ \dots & \dots & \dots & \end{pmatrix} \tag{2.2}$$

Then the stationary distribution $\{\pi_i\}$ satisfies:

$$\begin{cases} P_{1,1}\pi_1 + P_{2,1}\pi_2 + P_{3,1}\pi_3 + \cdots = \pi_1 \\ P_{1,2}\pi_1 + P_{2,2}\pi_2 + P_{3,2}\pi_3 + \cdots = \pi_2 \\ P_{1,3}\pi_1 + P_{2,3}\pi_2 + P_{3,3}\pi_3 + \cdots = \pi_3 \\ \qquad\qquad\qquad\qquad\qquad\qquad \dots \\ \pi_1 + \pi_2 + \pi_3 + \cdots = 1 \end{cases} \tag{2.3}$$

In the context of sampling process, π_i represents the probability that a node i is visited during the sampling process. When $\pi_i = \pi_j = \frac{1}{|V|}$, for $\forall i, j \in V$, the sampling algorithm is unbiased based on nodes. Thus, Eq. 2.4 can be got.

$$\begin{cases} P_{1,1} + P_{2,1} + P_{3,1} + \cdots = 1 \\ P_{1,2} + P_{2,2} + P_{3,2} + \cdots = 1 \\ P_{1,3} + P_{2,3} + P_{3,3} + \cdots = 1 \\ \qquad\qquad\qquad\qquad \dots \\ \pi_1 = \pi_2 = \pi_3 = \cdots = \frac{1}{|V|} \end{cases} \tag{2.4}$$

which completes the proof of necessity.

Let's then move to the proof of sufficiency. That is, if for $\forall i \in V$, $\sum_{j=1}^{|V|} P_{j,i} = 1$ is known, the work tries to deduce the sampling algorithm is unbiased. As stated in *Lemma 1*, the Markov chain abstracted from the sampling process on the graph is ergodic and irreducible. This implies that the Markov process is stationary and the stationary distribution is the unique solution for Eq. 2.3 [73]. A solution for the State Transition Probabilities equations Eq. 2.3 is: $\pi_i = \frac{1}{|V|}$, for $\forall i \in V$. This is in fact

also the unique solution. Under such a solution, the obtained samples are unbiased on node. This completes the proof of sufficiency. □

The work summarizes from *Theorem 1* two critical conditions for unbiased random walk-based sampling of networks having at least one node with non-zero clustering coefficient (which is a major feature of online social graphs):

(1) The moving probabilities between two nodes i and j should satisfy: if $P_{i,j} > 0$, then $P_{j,i} > 0$;

(2) $\forall i \in V, \sum_{j=1}^{|V|} P_{j,i} = 1$.

The exist of local disassortative nodes makes the design of an unbiased sampling method challenging. A high local disassortativity (*i.e.* ρ is negative with large absolute value) implies that a high-degree node i is surrounded by lots of low-degree nodes. The random crawler therefore arrives at i's low-degree neighbor j from i with a low probability, *i.e.* $P_{i,j}$ is low. In this context, two kinds of method could be adopted to meet the second unbiased condition on node j. One is to randomly choose a set of nodes which are not neighbors of j and allow the sampling crawler to jump from j to these nodes with certain probabilities [81][80]. However, such random jumps could lead to many small connected components sampled, instead of well-connected subgraphs. The other solution is to let the crawler stay on j rather than move to other nodes with certain probability once the crawler visits a low-degree node j, which is the intuition of MHRW [34].

2.3.3 Limitations of MHRW

Metropolis-Hastings Random Walk(MHRW) is proposed in [34] to obtain unbiased and well-connected samples of online social networks (like Facebook). The probability of moving from a node u to its neighbor v for the sampling crawler is computed as follows:

$$P_{u,v} = \begin{cases} min(\frac{1}{k_v}, \frac{1}{k_u}) & \text{if } v \text{ is a neighbor of } u \\ 1 - \sum_{y \neq u} P_{u,y} & \text{if } v = u \\ 0 & \text{otherwise} \end{cases} \tag{2.5}$$

It is easy to find that for any two adjacent nodes u and v, $P_{u,v} = P_{v,u} = min(\frac{1}{k_v}, \frac{1}{k_u}) > 0$. Besides, $\forall i \in V, \sum_{j=1}^{|V|} P_{j,i} = \sum_{j=1}^{|V|} P_{i,j} = 1$. Following *Theorem 1*, MHRW can be concluded as an unbiased sampling method.

MHRW achieves unbiased sampling at the cost of repetitively crawling low-degree nodes from themselves. The work denotes the moving probability of the sampling crawler from a node i to itself (*i.e.* $P_{i,i}$ in Eq. 2.5) as *self-sampling probability*. The self-sampling is likely to happen on low-degree nodes that are connected with high-degree ones, leading to local disassortativity.

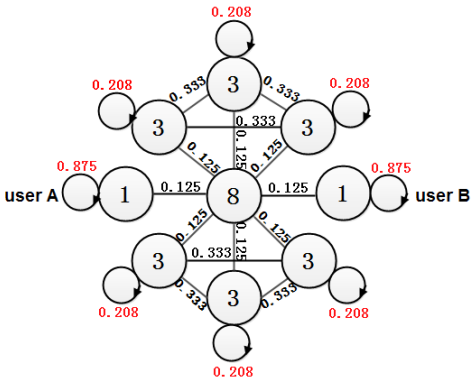


Figure 2-2: High self-sampling probability in MHRW

Fig. 2-2 illustrates the high self-sampling probability problem in MHRW. Node degrees are shown on the nodes and the transition probabilities are labeled on the edges. The work also depicts the self-sampling probabilities in MHRW beside the nodes. The local disassortativity for the central node is -0.2919, which shows a high local disassortativity compared with other nodes (0.0818 on the six 3-degree nodes and 0 on the two 1-degree nodes). In Fig. 2-2, the self-sampling probabilities of node A and B are as high as 0.875. Once the sampling crawler arrives at one of these two nodes, it will take a long time to sample the node itself repeatedly.

MHRW provides an unbiased estimation of topological properties of the network

(*e.g.* distribution of node degree, clustering coefficient) by taking a node sampled by m times as m separated nodes with the same properties, *i.e.* the average node degree is estimated as follows:

$$E_r(k) = \frac{\sum_{v \in V'} k_v N_v}{\sum_{v \in V'} N_v} \tag{2.6}$$

where V' is the set of unique sampled nodes, k_v is the degree of $v \in V'$ and N_v is the sampling times of v during the sampling process. As shown in [34], after several thousands iterations in Facebook, the estimated average node degree is close to the ground truth. However, sometimes researchers are interested in the sampled subgraph where the repetitions should be removed, *e.g.* to use the sampled subgraphs as the basis for analysis on individual user behavior. In these cases, a sampled node is only counted once independent of the times that it is sampled. Then, the average unique node degree of sampled subgraph is as follows:

$$E_u(k) = \frac{\sum_{v \in V'} k_v}{|V'|} \tag{2.7}$$

where V' is the set of unique sampled nodes and $|V'|$ is the number of unique sampled nodes.

In online social networks like Facebook, users are likely to connect to those with similar profiles (*e.g.* classmates). In this case, the self-sampling probabilities of individual nodes might be low and the difference between $E_r(k)$ and $E_u(k)$ is small. However, in online social media, there are plenty of local disassortative nodes of which neighbors have high self-sampling probabilities, leading to a much higher $E_u(k)$ compared with $E_r(k)$. A high self-sampling times per node also impairs the sampling efficiency of discovering new type of attributes. The results in the evaluation sections (Section 2.5 and 2.6) confirm these observations.

2.4 Unbiased Sampling with Dummy Edges

This section presents a novel unbiased sampling method, called *USDE*. The basic idea is to keep the connectivity and unbias nature of samples obtained by MHRW while avoid the excessive self-sampling probability. To this end, USDE adds dummy edges between low-degree nodes and amortizes self-sampling probabilities of individual nodes to moving probabilities on the dummy edges. It is worth noting that dummy edges are used only during sampling process to help the sampling crawler move to another node and they are not involved in the sampled subgraphs. In what follows, the paper first details the concept of *dummy edge* and describes the computation of moving probabilities in USDE, and then analyzes the sampling efficiency of USDE.

2.4.1 Dummy edges

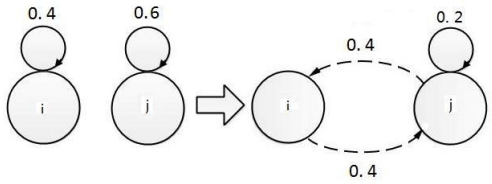


Figure 2-3: An equivalent case of two non-adjacent nodes with non-zero self-sampling probability

A simple example is shown in Fig. 2-3 to illustrate the usage of dummy edges. Node i and j are two non-adjacent nodes in a graph. Supposing in an unbiased assignment of moving probabilities (*e.g.* MHRW), the self-sampling probabilities of i and j are $L_i = 0.4$ and $L_j = 0.6$. USDE can add a bidirectional dummy edge between i and j and set $P_{i,j} = P_{j,i} = 0.4$, at the same time update $L_i = 0$ and $L_j = 0.2$. The self-sampling probabilities of i and j are reduced, while USDE still keeps the moving probabilities satisfying the unbiased conditions obtained in Section 2.3.

Analogously, USDE can add multiple dummy edges from one node. Supposing the crawler is currently on the node i, the candidate nodes for building dummy edges from i are the previously visited nodes' neighbors that have not been visited yet and have non-zero self-sampling probabilities. In this way, USDE avoids jumping to

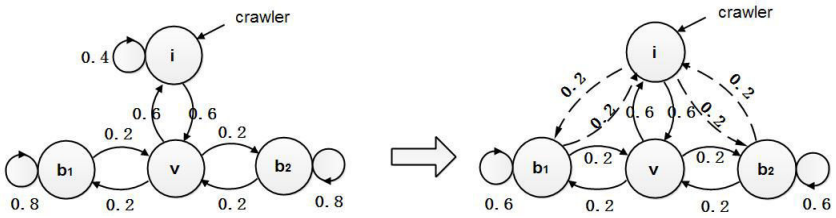


Figure 2-4: Illustrating example of adding dummy edges on multiple nodes

random nodes as in [81][80] which might result in many connected components in samples. USDE allows the crawler to obtain the ID and degree information of the i's neighbors when visiting a node i. This one-hop look-ahead operation is practical in social media through standard API calls. Besides, USDE stores the dummy edges information during sampling process. For a dummy edge between i and j, USDE stores a tuple $(i, j, DP_{i,j})$, where $DP_{i,j}$ is the moving probability of the dummy edge. This provides the crawler with the knowledge of dummy edges for individual unvisited nodes. Supposing the node ID is a digital number not exceeding 32 bits (as in Sina Weibo), each tuple requires 12 bytes storage space. As the designers of USDE always set an upper bound on a node's dummy edges in practice (*e.g.* 100 in this work), sampling as many as 10,000 iterations requires using only 12 Mbytes space at most.

Fig. 2-4 shows an example where USDE adds multiple dummy edges between nodes. The sampling crawler is at node i with 0.4 self-sampling probability. Two nodes b_1 and b_2 have not been visited yet and their self-sampling probabilities are $L_{b_1} = L_{b_2} = 0.8$. If USDE adds two dummy edges $\{i, b_1\}$ and $\{i, b_2\}$ and assigns the moving probabilities on the two edges as $P_{i,b_1} = P_{b_1,i} = 0.2$, $P_{i,b_2} = P_{b_2,i} = 0.2$, then L_i, L_{b_1}, L_{b_2} are reduced to 0, 0.6 and 0.6 respectively. As b_1 and b_2 have not been visited yet, moving the crawler to these nodes would improve the efficiency of identifying new nodes and new user attributes.

Indeed before visiting a node, the knowledge of its exact self-sampling probability is not available. USDE thus proposes to estimate the lower bound of the self-sampling probabilities for neighbors of visited nodes using only the neighbors' degree information, which will be detailed in the next subsection.

2.4.2 Moving Probability in USDE

Since USDE focuses on reducing the self-sampling probabilities of individual nodes in MHRW, USDE adopts the moving probability from a node i to its neighbor v in MHRW. That said, $P_{i,v}$ and $P_{v,i}$ are computed as $min(\frac{1}{k_v}, \frac{1}{k_i})$, where k_i and k_v are degrees of i and v, respectively. Supposing there is a dummy edge between node i and node j, the paper denotes the moving probability from i to j as $DP_{i,j}$ and sets $DP_{i,j} = DP_{j,i}$.

Let $U(i)$ denote the set of nodes which have dummy edges with i, $S(i)$ denote the set of neighbors of i. The moving probability for the sampling crawler from node i to node v in USDE is computed as follows:

$$P_{i,v} = \begin{cases} min(\frac{1}{k_v}, \frac{1}{k_i}) & \text{if } v \in S(i) \\ DP_{i,v}, & \text{if } v \in U(i) \\ 1 - \sum_{x \in S(i)} P_{i,x} - \sum_{y \in U(i)} P_{i,y} & \text{if } v = i \\ 0 & \text{otherwise} \end{cases} \tag{2.8}$$

If $DP_{i,v} = 0$, $\forall v \in U(i)$, then USDE is identical to MHRW. It can be found from the Eq. 2.8 that $P_{i,j} = P_{j,i}$. Thus, if $P_{i,j} > 0$, then $P_{j,i} > 0$. Besides, $\forall i \in V, \sum_{j=1}^{|V|} P_{j,i} = \sum_{j=1}^{|V|} P_{i,j} = 1$, where $|V|$ is the number of nodes in the graph for sampling. According to *Theorem 1*, USDE could obtain unbiased samples.

The paper then describes the selection of nodes to build dummy edges and the computation of moving probabilities on dummy edges. The nodes i to which USDE could build dummy edges from the currently visited node are the previously visited nodes' neighbors that should have a non-zero self-sampling probability and have not been visited by the sampling crawler. However, finding such nodes is challenging because the crawler could not get the exact moving probability from i to its neighbors as it has not been visited yet. In this context, USDE proposes to estimate the lower bound of i's self-sampling probability during sampling process.

It can be found from Eq. 2.8 that if there is a neighbor u for node i with degree $k_u > k_i$, then the self-sampling probability without dummy edges of i is at least

$(\frac{1}{k_i} - \frac{1}{k_u})$. Using this observation, the lower bound of self-sampling probability without dummy edges of an unvisited node i, LP_i, can be derived as follows:

$$LP_i = \sum_{v \in \{S(i) \cap V'\}} (\frac{1}{k_i} - \frac{1}{k_v}) \quad \text{where } k_v > k_i \tag{2.9}$$

where $S(i)$ is the neighbor set of i and V' is the set of nodes that have been visited.

Fig. 2-5 illustrates how the lower bounds are estimated during the sampling process. The node degrees are marked on the nodes and the node IDs are labeled beside the nodes. Let's assume that for all nodes, $LP = 0$ before step T. At step T, the crawler visits node w. Node j and node m are neighbors of w and $k_j < k_w$, $k_m < k_w$. Hence, it can be estimated that $LP_j = \frac{1}{k_j} - \frac{1}{k_w} = \frac{1}{20} - \frac{1}{50} = 0.03$ and $LP_m = \frac{1}{k_m} - \frac{1}{k_w} = \frac{1}{10} - \frac{1}{50} = 0.08$ at step T. At the next step $T+1$, the crawler visits y. USDE updates the estimation of LP_j because j is also a neighbor of y and $k_j < k_y$. The LP value of j is updated as $LP_j = 0.03 + \frac{1}{20} - \frac{1}{30} = 0.0467$. Note that USDE cannot estimate LP_n at $T+1$ as $k_n > k_y$.

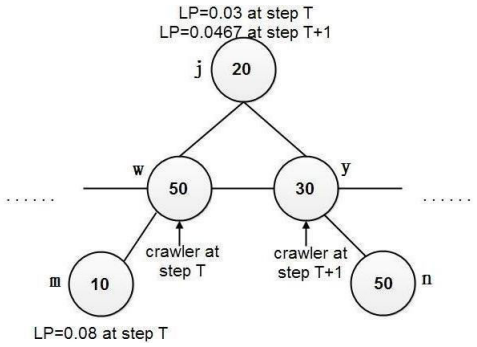


Figure 2-5: LP estimations during the process of USDE

With the estimated lower-bounds of self-sampling probabilities for unvisited neighbors of the sampled nodes, USDE now can assign the moving probability to individual dummy edges. During the process of sampling, USDE uses a queue Q to record the node ID and the estimated lower-bound of self-sampling probability LP_v for each unvisited node v with $LP_v > 0$, *i.e.* a tuple (v, LP_v) for a node v. When a node i is visited for the first time, USDE can obtain its current self-sampling probability with $1 - \sum_{x \in S(i)} P_{i,x} - \sum_{y \in U(i)} P_{i,y}$ according to Eq. 2.8. $U(i)$ is empty in the case

that i has never been selected for building dummy edges before. If such a probability is larger than a threshold T, USDE pops a tuple (v, LP_v) from Q and add a new dummy edge between nodes i and v. The moving probability is computed as $DP_{i,v} = DP_{v,i} = min(LP_v, 1 - \sum_{x \in S(i)} P_{i,x} - \sum_{y \in U(i)} P_{i,y})$. Then, USDE updates LP_v with $LP_v^{'} = LP_v - DP_{v,i}$. If the updated $LP_v^{'} > 0$, USDE pushes the updated tuple $(v, LP_v^{'})$ back to Q. Such an estimation method for the lower-bound of self-sampling probability ensures that the sampled subgraphs are well-connected because all the nodes recorded in Q are neighbors of sampled ones.

The moving probability $DP_{i,v}$ ($DP_{v,i}$) on the dummy edge between v and i might be high if both LP_v and the self-sampling probability without dummy edges on i are high. In this case, the sampling crawler would wander between nodes i and v for a long time, which prevents the crawler from finding new nodes. To solve this problem, rather than pop only 1 tuple, USDE pops γ ($\gamma > 1$) tuples (v_1, LP_{v_1}), (v_2, LP_{v_2}),(v_3, LP_{v_3}) $\cdots$ $(v_\gamma, LP_{v_\gamma})$ from Q at one time, and γ dummy edges are added $\{i,v_j\}$ $(j = 1, 2 \cdots \gamma)$. The moving probability on dummy edge $\{i, v_j\}$ is then computed as $DP_{i,v_j} = DP_{v_j,i} = min(LP_{v_j}, \frac{1 - \sum_{x \in S(i)} P_{i,x} - \sum_{y \in U(i)} P_{i,y}}{\gamma})$. Then LP_{v_j} is updated accordingly and the tuples with non-zero LP are pushed back to Q. The addition of dummy edges stops when the self-sampling probability on i is reduced to 0 or dummy edges to all the γ nodes are added.

The queue Q applies FIFO (First In, First Out), which makes the unused nodes likely to be popped. It avoids adding too many dummy edges on a single node. Moreover, the LP of a node will become smaller as more dummy edges are built to it. With FIFO, the stored nodes with fewer dummy edges and higher LP are more likely to be popped, which makes sure that self-sampling probability can be reduced as much as possible. The queue Q is initialized in the first t iterations of sampling. During this time period, dummy edges are not added but the values of LP of the visited nodes' neighbors are estimated and pushed into Q.

2.4.3 Implementation of USDE

The pseudo code of USDE for online social media is listed as follows, where t (the number of iterations for the queue initialization), T (the threshold on self-sampling probability) and γ (the number of records popped from Q at a time) are three design parameters. The list R stores the sampled nodes uniquely.

In the algorithm, `USDE` is the main procedure, `EstimateLP` is the procedure for estimation of the lower bound of self-sampling probability, and `UpdateProbability` is the procedure for the update of self-sampling probability after adding dummy edges. Each node i is associated with a hash table U_i to store the information of dummy edges of which it is an end point. In detail, a tuple $(j, DP_{i,j})$ is stored in U_i for the dummy edge $\{i, j\}$. It is also worth noting that once node i is visited by the sampling crawler, it should not be used as a candidate for building dummy edges from others, simply because its self-sampling probability has been reduced as much as possible. That said, (i,LP_i) should be removed from Q when i is visited (see line $7 \sim 8$ in the procedure `USDE`).

```
    UpdateProbability(v,Q,γ)
1   N ← 0
2   while N < γ and L_v > 0
3         do pop (b, LP_b) from Q
4            P_{v,b} = P_{b,v} ← min(LP_b, L_v/γ)
5            if U_v, U_b don't exist
6               then U_v ← ∅, U_b ← ∅
7            push(b, P_{v,b}) into U_v, push (v, P_{b,v}) into U_b
8            L_v ← L_v − P_{v,b}, LP_b ← LP_b − P_{b,v}
9            if LP_b > 0
10              then push (b,LP_b) into Q
11           N ← N + 1
```

```
    ESTIMATELP(v,i,Q,R)
1   if k_v > k_i and i ∉ R and i ∈ Q
2       then update LP_i ← LP_i + (1/k_i − 1/k_v) in Q
3   if k_v > k_i and i ∉ R and i ∉ Q
4       then LP_i ← 1/k_i − 1/k_v, push (i,LP_i) into Q

     USDE(t,T,γ)
 1  R ← ∅, Q ← ∅, Iteration ← 1, v ← inital node
 2  while stopping criterion has not been met
 3        do if v ∉ R
 4              then push v in R
 5                   if U_v doesn't exist
 6                      then U_v ← ∅
 7                   if (v, LP_v) ∈ Q
 8                      then pop (v, LP_v) from Q
 9                   L_v ← 1
10                   for each neighbor i of v
11                        do P_{i,v} = P_{v,i} ← min(1/k_i, 1/k_v)
12                           L_v ← L_v − P_{v,i}
13                           EstimateLP(v, i, Q, R)
14                   for each tuple (k, P_{v,k}) ∈ U_v
15                        do L_v ← L_v − P_{v,k}
16                   if Iteration > t and L_v > T
17                      then UpdateProbability(v, Q, γ)
18           Select a node w according to the probability distribution {P_{v,w}}
19           v ← w
20           Iteration ← Iteration + 1
```

USDE leverages multiple random-walk based crawlers that run in parallel for efficient sampling. At the beginning, a set of initial nodes (called seeds) are uniformly selected from the network at random. Each seed initializes a crawler following the

above sampling algorithm. Crawlers share the common queue Q of candidates of end points for dummy edges (the queue Q is associated with a mutex lock). In this way, each crawler is able to access the candidate end points for dummy edges generated by other crawlers, enabling the building of dummy edges between two nodes in different communities of the network. Finally, nodes that are visited by crawlers, along with the edges between them, are exported to form the final sampled subgraph. The dummy edges however are not included in the final result. As shown in the evaluation (Section 2.5 and Section 2.6), using multiple crawlers generates unbiased and well-connected samples.

A practical concern is how the initial seeds can be chosen uniformly at random in an efficient way. In online social medias, users are often numerically identified and user profiles can be accessed using the IDs. A lot of online social medias (like Twitter, Sina Weibo) have dense ID space. User IDs thus can be uniformly generated at random and be used as initial seeds. However, some social medias, like Google Plus, do use sparse ID space. Fortunately, popular social medias always provide public profile directories that can be used for random seeds selection. For example, Google maintains a sitemap file that contains a link to every Google Plus profile public[1]; Twitter[2] and Facebook[3] also provide public user profile directories. The initial seeds can be uniformly chosen from the users listed in the profile directories at random as in [24]. Using either randomly generated IDs or public profile directory is able to pick initial seeds efficiently, as not more than 100 seeds are required in this work.

2.4.4 Analysis of Sampling Efficiency of USDE

The work analyzes the sampling efficiency of USDE from two perspectives: the speed in discovering new nodes during the sampling process and the convergence of crawlers. This analysis shows that the addition of dummy edges in USDE can reduce the sampling times per node and improve the convergence by increasing the "conductance"

[1]http://www.gstatic.com/s2/sitemaps/profiles-sitemap.xml
[2]https://twitter.com/i/directory/profiles
[3]http://www.facebook.com/directory

of the sampled network, compared with MHRW.

Sampling times per node

Given the sampling iterations, the speed in discovering new nodes is inversely proportional to the sampling times per node. Assuming the crawler starts from a node i, the average sampling times on node i within T steps are $\sum_{n=1}^{T} P_{i,i}^{(n)}$. The n-step transition probability $P_{i,i}^{(n)}$ includes two parts: the probability of the crawler staying at i for n iterations continuously and the sum of probabilities that the crawler staying at i for $(n-m-2)$ iterations continuously and then making a $(m+2)$-step loop through a pair of attached edges of i, where $0 \leq m \leq (n-2)$. Therefore, $P_{i,i}^{(n)}$ can be derived as follows:

$$P_{i,i}^{(n)} = P_{i,i}^{n} + \sum_{m=0}^{n-2} P_{i,i}^{n-2-m} \sum_{\forall j,k \in V} P_{j,i} P_{k,i} P_{j,k}^{(m)} \tag{2.10}$$

where $P_{i,i}$ is the self-sampling probability of i. It is worth noting that only the adjacent nodes of i can be considered in Eq. 2.10 because for any nonadjacent node j, $P_{i,j} = P_{j,i} = 0$. It is hard to get the exact value of $P_{i,i}^{(n)}$ because given m, $P_{j,k}^{(m)}$ varies for different node pairs (j,k). However, as stated in Section 2.3, the sampling process is stationary and $\forall j,k, \pi_j = \pi_k = \frac{1}{|V|}$. That said, when $P_{j,k}^{(m)}$ reaches the stable state, its value is equal to π_j $(i.e. \frac{1}{|V|})$ for $\forall j,k \in V$. To ease analysis, this paper assumes $\forall j,k \in V, P_{j,k}^{(m)}$ is in the stationary state and thus is close to $\frac{1}{|V|}$. Then Eq. 2.10 can be written as:

$$\begin{aligned}
P_{i,i}^{(n)} &= P_{i,i}^{n} + \sum_{m=0}^{n-2} P_{i,i}^{n-2-m} \sum_{\forall j,k \in S(i)} P_{j,i} P_{k,i} \frac{1}{|V|} \\
&= P_{i,i}^{n} + \sum_{m=0}^{n-2} P_{i,i}^{n-2-m} \frac{1}{|V|} \sum_{\forall j \in S(i)} P_{j,i} \sum_{\forall k \in S(i)} P_{k,i} \\
&= P_{i,i}^{n} + \sum_{m=0}^{n-2} P_{i,i}^{n-2-m} \frac{1}{|V|} (1 - P_{i,i})^2 \\
&= P_{i,i}^{n} + \frac{(1 - P_{i,i})(1 - P_{i,i}^{n-1})}{|V|}
\end{aligned} \tag{2.11}$$

With the dummy edges on node i in USDE, the self-sampling probability of i is reduced from $P_{i,i}$ to $\hat{P}_{i,i}$. The n-step moving probability of i in USDE thus can be obtained as follows:

$$\begin{aligned}
\hat{P}_{i,i}^{(n)} &= \hat{P}_{i,i}^{n} + \sum_{m=0}^{n-2} \hat{P}_{i,i}^{n-2-m} \sum_{\forall j,k \in S(i)} P_{j,i} P_{k,i} P_{j,k}^{(m)} \\
&+ \sum_{m=0}^{n-2} \hat{P}_{i,i}^{n-2-m} \sum_{\forall h,l \in U(i)} P_{h,i} P_{l,i} P_{h,l}^{(m)} \\
&+ 2 \sum_{m=0}^{n-2} \hat{P}_{i,i}^{n-2-m} \sum_{\forall j \in S(i), \forall l \in U(i)} P_{j,i} P_{l,i} P_{j,l}^{(m)} \\
&= \hat{P}_{i,i}^{n} + \sum_{m=0}^{n-2} \hat{P}_{i,i}^{n-2-m} \frac{1}{|V|} (1 - \hat{P}_{i,i})^2 \\
&= \hat{P}_{i,i}^{n} + \frac{(1 - \hat{P}_{i,i})(1 - \hat{P}_{i,i}^{n-1})}{|V|}
\end{aligned} \tag{2.12}$$

where $S(i)$ is the neighbor set of i (excluding the dummy edges) and $U(i)$ is the end-point set of i's dummy edges.

Now $\sum_{n=1}^{T} P_{i,i}^{(n)}$ and $\sum_{n=1}^{T} \hat{P}_{i,i}^{(n)}$ for a finite T can be compared. According to Eq. 2.11 and Eq. 2.12, $\sum_{n=1}^{T} P_{i,i}^{(n)}$ can be rewritten as $\frac{P_{i,i}(1-P_{i,i}^{T})}{1-P_{i,i}} + \sum_{n} \frac{(1-P_{i,i})(1-P_{i,i}^{T-1})}{|V|}$. In large-scale social media networks, $|V|$ is a large number (*e.g.* in Twitter, $|V|$ is over 200 million). That said, $\frac{P_{i,i}(1-P_{i,i}^{T})}{1-P_{i,i}}$ is likely to be dominant in $\sum_{n=1}^{T} P_{i,i}^{(n)}$ if T is finite and $P_{i,i}$ is not low. For example, supposing $P_{i,i} = 0.2$ and $|V| = 1,000,000$, $\frac{P_{i,i}(1-P_{i,i}^{T})}{1-P_{i,i}} = 0.251$ while $\sum_{n} \frac{(1-P_{i,i})(1-P_{i,i}^{T-1})}{|V|} = 0.008$ even for $T = 10,000$. Indeed, $\frac{P_{i,i}(1-P_{i,i}^{T})}{1-P_{i,i}}$ is a monotone increasing function over $P_{i,i}$. In other words, a reduced self-sampling probability leads to reduced average sampling times on nodes. As adding dummy edges can efficiently reduce the self-sampling probabilities on individual nodes, it is easy to conclude that USDE can improve the speed in discovering new nodes.

Convergence of USDE

In graph theory, the "conductance" of a graph $G(V, E)$ measures how "well-knit" the graph is and it controls how fast a random walk-based crawler on G converges

to a uniform distribution of sampling probabilities[14]. This work thus studies the convergence of USDE from the perspective of conductance of sampled graphs. The conductance of a set S in graph G is defined as follows[50]:

$$\varphi(S) = \frac{F(S)}{\pi(S)(1-\pi(S))} \tag{2.13}$$

where $F(S)$ is called as the *ergodic flow* from S and defined as $F(S) = \sum_{i \in S} \sum_{j \notin S} \pi_i P_{i,j}$. $F(S)$ denotes the fraction of steps of a very long random walk which move from S to its complement. The denominator denotes the fraction of steps of a very long sequence of independent samples from π which move from S to its complement, where $\pi(S) = \sum_{i \in S} \pi_i$. The conductance of the whole graph is the minimum conductance over all nonempty proper subsets of V.

A small conductance always indicates that the network consists of several communities which have few links to others, rather than a whole connected graph. In this context, USDE with a single random walk crawler may prevent the crawler in one community from discovering other communities as the dummy edges are always added in the community of the starting node.

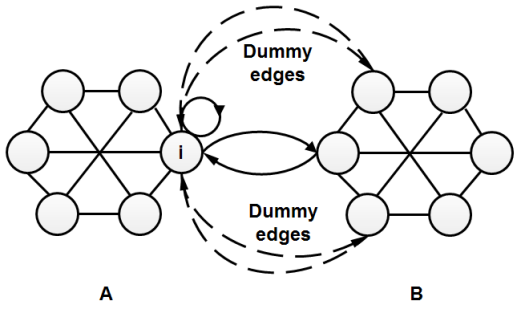


Figure 2-6: USDE with multiple crawlers in a barbell graph

However, similar to MHRW, USDE is also a MCMC technology where multiple running crawlers are necessary. At the beginning, a set of initial nodes are uniformly selected from the network at random, each of which corresponds to a random walk crawler. All crawlers share the common queue Q of candidates of end points for dummy edges[4]. Following this way, crawlers in one community can have access to

[4]The queue Q is associated with a mutex lock.

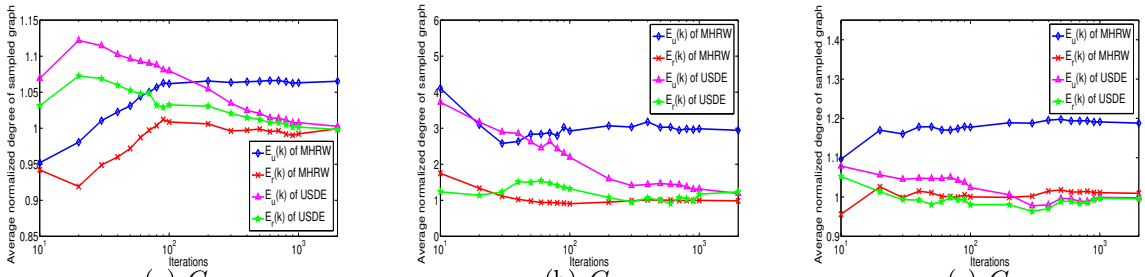
the candidates of end points for dummy edges in other communities, which enables the building of dummy edges between two nodes in different communities, as shown in Fig. 2-6. Although in the multiple version, one crawler may be affected by the extra dummy edges of the other crawlers, all additions of dummy edges follow the principle that the values of diagonal entries of transition matrix are assigned to the ones in the same columns and the same rows as much as possible while keeping the sum of values in each row and each column are 1. Therefore, the additions of dummy edges concurrently do not break the unbiased sampling conditions derived in Section 2.3 for each crawler and the final samples are still unbiased.

As depicted in Section 2.4.2, USDE remains the moving probability of MHRW from one node to its neighbors following the real relationships and the stationary distribution $\hat{\pi}_i = \hat{\pi}_j = \frac{1}{|V|}, \forall i, j \in V$ which is also same as the one in MHRW. This paper uses $\varphi(S)$ to be the minimum conductance of G in MHRW and $F(S)$ to be the ergodic flow from the community S, besides, $D(S)$ is defined to be the set of dummy edges between nodes $i \in S$ and nodes $j \notin S$ added by USDE. Then the new conductance of S in USDE $\hat{\varphi}(S)$ is as follows:

$$\begin{aligned} \hat{\varphi}(S) &= \frac{\hat{F}(S)}{\hat{\pi}(S)(1-\hat{\pi}(S))} \\ &= \frac{F(S) + \sum_{i \in S} \sum_{(i,j) \in D(S)} \hat{\pi}_i P_{i,j}}{\pi(S)(1-\pi(S))} \\ &= \varphi(S) + \frac{\sum_{i \in S} \sum_{(i,j) \in D(S)} \pi_i P_{i,j}}{\pi(S)(1-\pi(S))} \end{aligned} \tag{2.14}$$

Obviously, $\hat{\varphi}(S)$ can be larger than $\varphi(S)$ once $D(S)$ is nonempty. In other words, if there are dummy edges added by USDE between S and its complement, it can increase the "conductance" of the graph and therefore improve the convergence of sampling compared with MHRW.

The condition for $D(S)$ being non-empty is that some crawlers start from S *and* some from the complement of S. Suppose the proportion of nodes in S is ρ_S. Given that the initial seeds are randomly selected, the probability of $D(S)$ is non-empty is $p_{D(S)} = 1 - \rho_S^n - (1-\rho_S)^n$, where n is the number of crawlers running in parallel.



(a) G_g (b) G_p (c) G_b

Figure 2-7: Average normalized node degree sampled by USDE and MHRW

Therefore, $p_{D(S)}$ is very high with a limited number of crawlers. For instance, if $\rho_S = 5\%$, $p_{D(S)}$ is as high as 87% with only $n = 40$ crawlers (similar to MHRW in [35]).

2.5 Evaluation on Synthetic Networks

This section evaluates the performance of USDE on three synthetic networks, each with 20,000 nodes. Two of the three networks are generated using the *Exploration without replacement* algorithm [55] with a mean degree around 30: one with Gaussian distribution as input (referred as G_g) and the other with power-law distribution, which is close to the degree distribution of real-life online social media [58], as the input (referred as G_p). These two networks are used to measure the effect of local diassortativity. The third one is a barbell graph (referred as G_b), of which two communities with 10,000 nodes are generated with the Gaussian degree distribution and with mean degrees as 20 and 40 respectively. The work then randomly chooses from each community a node and links them with a single edge. This graph is used to measure how the conductance of graph controls the convergence rate of sampling. This work analyzes the local disassortativity for the first two graphs and finds that G_p shows a more significant local disassortativity than G_g as expected (as depicted in Fig. 2-8).

Besides the node degree, the work also associates nodes with user attributes. In particular, an integer $a \in [1, 200]$ is assigned to each node with different probability distributions (Gaussian and power-law) as the values for user attribute, which are

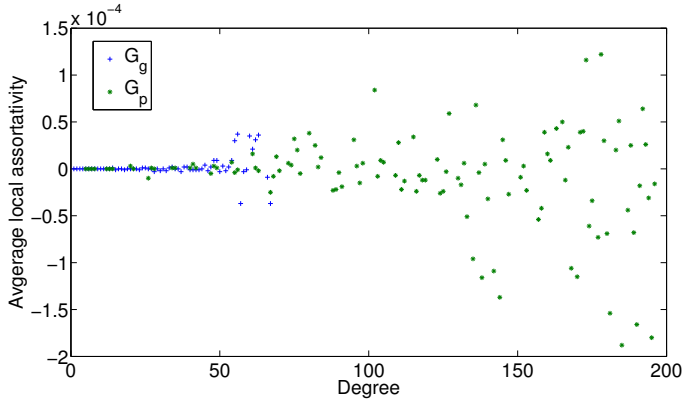


Figure 2-8: The average local assortativity vs. degree in synthetic networks with different degree distributions

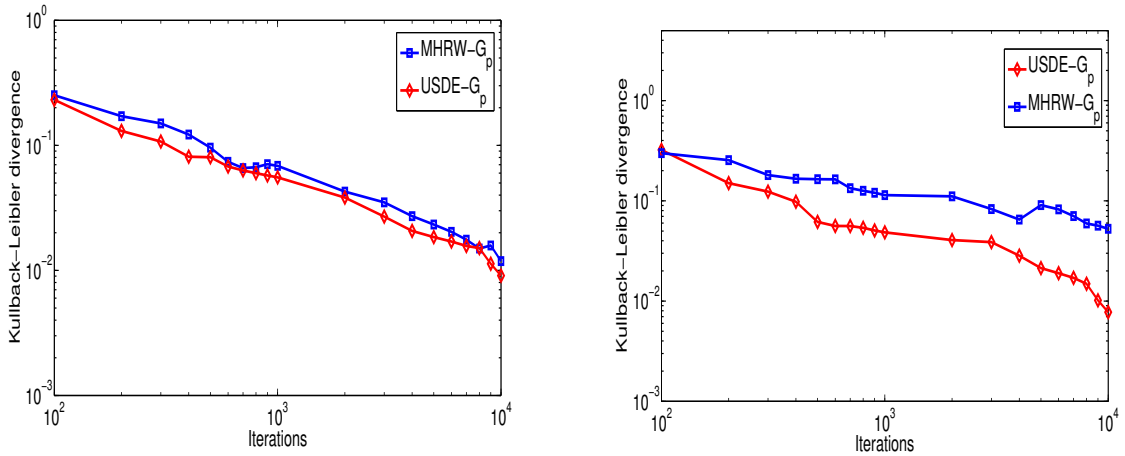
denoted as G-attribute and P-attribute respectively. The work runs sampling experiments with 10 crawlers simultaneously on each of the three graphs. Each experiment starts with a set of randomly chosen seeds. The default design parameters of USDE are set as: $t = 100$; $T = 0.00001$; $\gamma = 100$.

The work evaluates the performance from two perspectives: the quality of samples and the sampling efficiency. The quality of samples is measured by the closeness of sampled node degrees and user attributes to the values in ground truth, as well as the connectivity of the sampled subgraphs. The sampling efficiency on the other hand is measured by the convergence rate of crawlers on node degree, the average sampling times per node and the speed in identifying new user attributes.

2.5.1 Quality of samples

Fig. 2-7 plots the mean value of $E_r(k)$ (Eq. 2.6, average node degree *with* repetitions) and mean value of $E_u(k)$ (Eq. 2.7, average node degree *without* repetitions) of the 10 crawlers normalized by the ground truth of average node degree. Three observations are notable. First, both USDE and MHRW provide stable results after 300 iterations. Second, MHRW converges to a higher $E_u(k)$ than the ground truth, although $E_r(k)$ is close to the ground truth. On the other hand, both $E_u(k)$ and $E_r(k)$ of USDE converge to the ground truth. Specially, $E_u(k)$ of USDE approaches to the ground truth quickly after 100 iterations when the addition of dummy edges begins ($t = 100$). Finally, the tendency of local disassortative mixing pattern has a great impact on

$E_u(k)$ of MHRW, while it has limited impact on that of USDE. For example, the $E_u(k)$ of MHRW is 1.05 times as high as ground truth in G_g while 3 times in G_p.



(a) $G-attribute$ (b) $P-attribute$

Figure 2-9: The K-L divergence between distribution of sampled user attributes and the ground truth on G_p

The paper next examines the unbias of user attributes by computing the Kullback-Leibler (K-L) divergence between the distribution of sampled user attributes P and the distribution of ground truth Q. The K-L divergence between discrete probability distributions P and Q is computed as follows [53]:

$$D_{KL}(P||Q) = \sum_i ln(\frac{P(i)}{Q(i)})P(i) \tag{2.15}$$

$D_{KL}(P||Q)$ can be 0 if and only if $P = Q$ almost everywhere and the more significant the difference between P and Q is, the larger $D_{KL}(P||Q)$ becomes.

Fig. 2-9 shows the K-L divergence between the distribution of sampled user attributes by the two algorithms and the ground truth on G_p. It can be observed that the distribution of user attributes sampled by USDE is closer to the distribution of ground truth than that sampled by MHRW. In fact, the more biased the distribution of user attributes is, the more notable the advantage of USDE is. For example, for G-attribute, the K-L divergence of USDE is 0.004 less than that of MHRW in G_p after 10,000 iterations, while the difference grows to 0.07 for P-attribute. A detailed analysis reveals that MHRW misses the attributes which account for less than 0.1%

in ground truth while USDE indeed captures most of them.

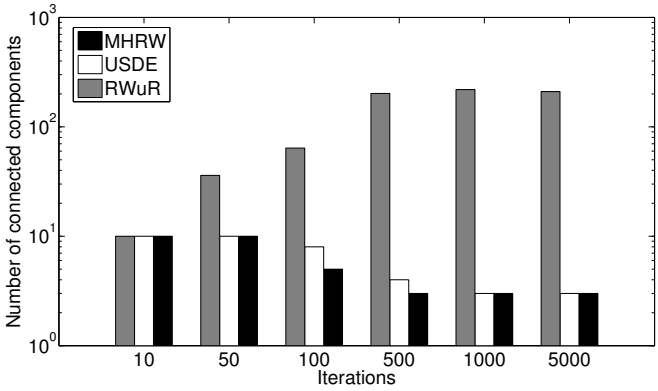
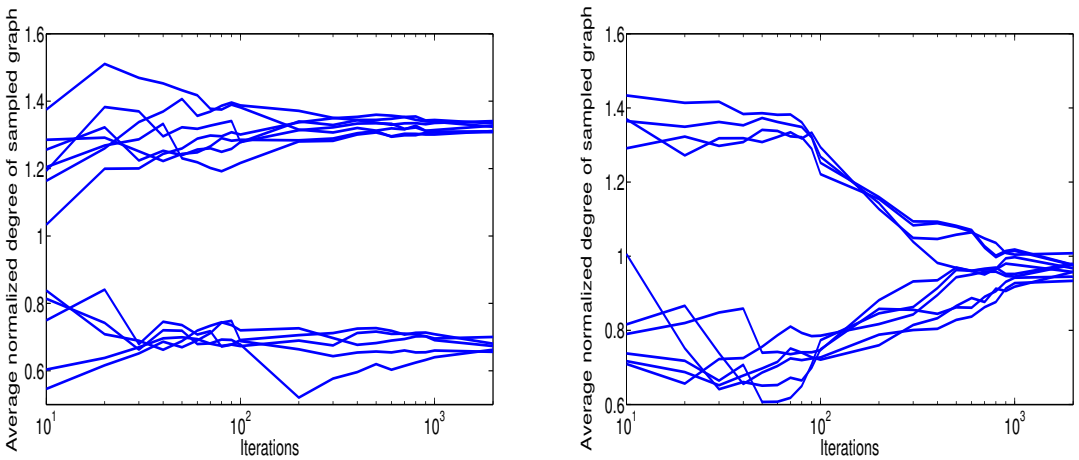


Figure 2-10: Number of connected components in samples generated by MHRW, USDE and RWuR on G_b

The work then measures the connectivity of samples by counting the number of connected components in samples based on the edges of complete networks, an important metric for the studies of user interactions and information propagation [87][92]. Besides MHRW, the work also compares USDE with RWuR on undirected graphs [8], which introduces the operation of random jumps in random walk to amend the bias. In RWuR, the crawler jumps to a randomly selected node from the current node i with the probability $\frac{\alpha}{k_i+\alpha}$, where α is the jumping weight and is set as 10 here (similar to [8]). Fig. 2-10 shows the number of connected components in samples generated by 10 crawlers of MHRW, USDE and RWuR on G_b. All the 10 crawlers of these three algorithms are not overlapped at the beginning of sampling process. The connectivity of samples of MHRW and USDE increases with the growth of iterations, while the one of RWuR decreases quickly. For example, with 5,000 iterations, USDE and MHRW generate only 3 connected components, while RWuR generates 200 connected components due to the random jumps. The results demonstrate that USDE can keep the connectivity of samples as in MHRW.

2.5.2 Sampling Efficiency

The work first measures the time required to obtain stable and accurate results for a crawler on the barbell network G_b. Fig.2-11 depicts the sampling convergence of



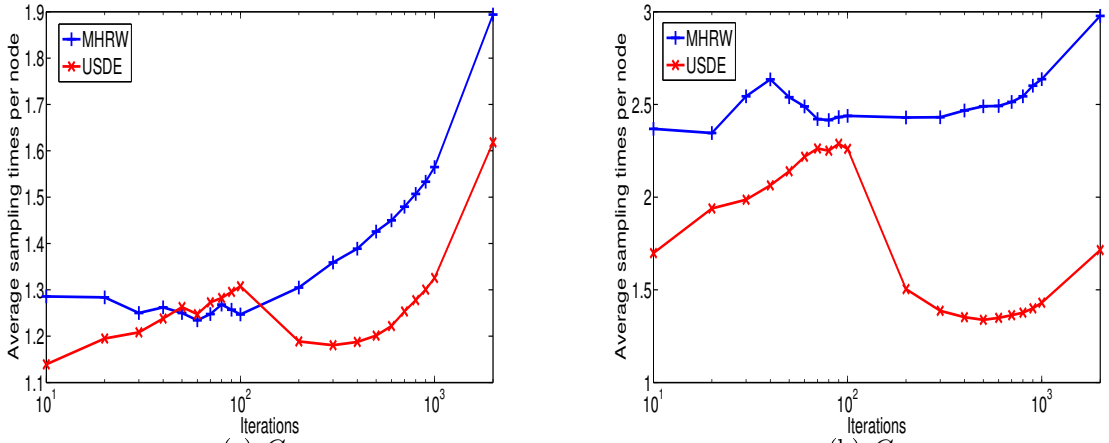
(a) MHRW (b) USDE

Figure 2-11: Convergence of MHRW and USDE on G_b

MHRW and USDE. It is notable that USDE has a better convergence than MHRW on such a network as analyzed in Section 2.4.4. For example, the sampled average degrees of all 10 crawlers in USDE are only 10% deviating from the ground truth after 600 iterations. In contract, all crawlers of MHRW cannot escape from the starting communities even after 2,000 iterations. Notably, all crawlers in USDE become convergent after 100 iterations when the addition of dummy edges starts.

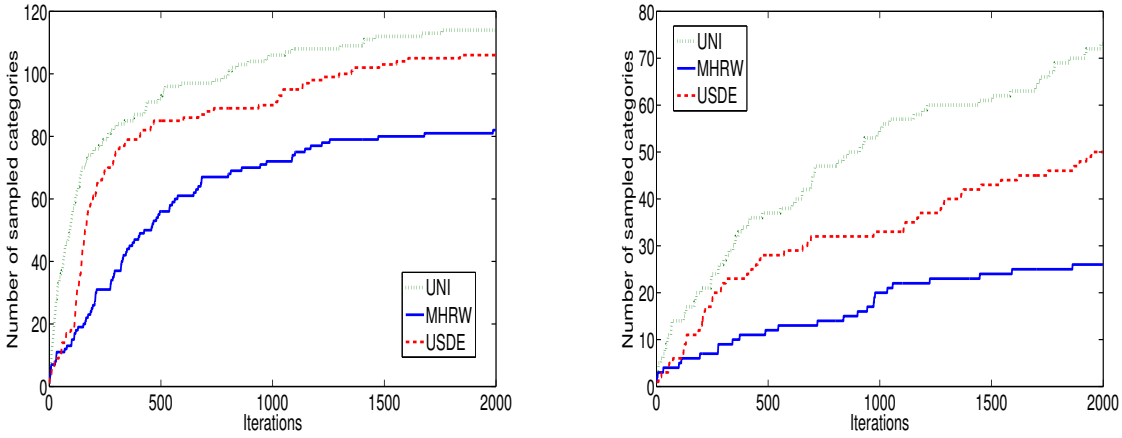
The work then evaluates the average sampling times per node during the sampling process on G_g and G_p in Fig. 2-12. It can be observed that sampling crawlers in MHRW revisit nodes for more times in the network where local disassortative mixing pattern is more notable. Within 1,000 iterations, the average sampling times per node of MHRW is 2.7 in G_p, greatly larger than that in G_g, 1.6. On the other hand, the average sampling times per node of USDE is much smaller than that of MHRW in any network. For example, the average sampling times per node of USDE is only 1.3 in G_p within 1,000 iterations. In particular, the difference of the average sampling times between the two methods becomes more significant after the addition of dummy edges in USDE starts (after 100 iterations).

The work finally examines the speed in identifying new attributes in Fig. 2-13, where the performance of UNI is used as baseline. USDE has a much closer performance to UNI than MHRW, indicating higher efficiency. For example, in Fig. 2-



(a) G_g (b) G_p

Figure 2-12: Average sampling times per node on G_g and G_p



(a) $G-attribute$ (b) $P-attribute$

Figure 2-13: Efficiency of identifying new attributes on G_p

13(a), while about 90 categories have been discovered by USDE within 1,000 sampling iterations, MHRW has discovered only 60 categories within the same period, implying USDE is 1.5 times as efficient as MHRW. The effect of user attribute distribution on the efficiency in identifying new attributes is also observed. It requires more sampling iterations to discover the same amount of categories for P-attribute than G-attribute because the more "slight" attributes (which account for small proportions), the harder crawlers discover them.

2.6 Sampling Twitter and Sina Weibo

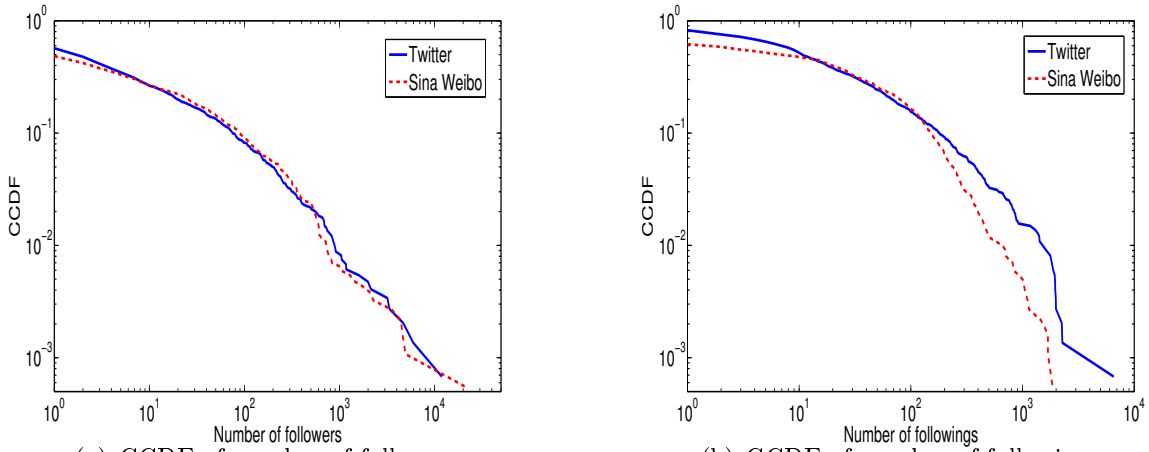
This section applies USDE to sample Twitter (the world's largest Microblog service) and Sina Weibo (a Chinese alternative of Twitter) in order to investigate the performance of USDE in the real world. Here the quality of samples and the sampling efficiency are also focused on as in Section 2.5.

2.6.1 Experiment Setup

In Twitter and Sina Weibo, the relationship between users might be non-reciprocal. USDE leverages the idea of "backward edge traversals" [104][81][80] to sample these two directed networks, where unidirectional edges are treated as bidirectional ones. In other words, the work considers for each node all its in-edges (follower relationships) and out-edges (following relationships) as undirected ones.

Given the huge size, one cannot get the real distributions of node degrees and user attributes of the whole Twitter or Sine Weibo networks. Fortunately, users in both Twitter and Sina Weibo are numerically identified by unique IDs. Thus, user IDs is generated uniformly at random and these IDs are used as input to query Twitter and Sina Weibo APIs. This procedure is indeed a UNI sampling method and could provide the ground truth. Both Twitter and Sina Weibo provide according APIs for third parties to collect the neighbors' information of a user given that user's ID: Twitter's API returns at most 200 neighbors' information for each call, while Sina Weibo's API returns at most 100 neighbors' information. The work runs the sampling methods using 10 crawlers simultaneously with different initial seeds and the average values of the 10 crawlers are reported.

Fig. 2-14 first plots the CCDF (Complimentary Cumulative Distribution Function) of the users' followers and followings in Twitter and Sina Weibo obtained by the above UNI sampling method. Neither Twitter nor Sina Weibo has a strict power-law distribution, which is in accordance with the reports in [4][58][69]. It can be also found that although the two curves for followers are similar, the distribution for followings in Twitter has a longer tail than the one in Sina Weibo, meaning that compared with



(a) CCDF of number of followers (b) CCDF of number of followings

Figure 2-14: Distribution for followers and followings in Twitter and Sina Weibo

Sina Weibo, there are more users with large number of followings in Twitter. The reason for this difference is that the current upper limit on the number of followings per user in Sina Weibo is 2,000, which was also the upper limit in Twitter before 2009 but is removed now[58]. Besides, more than 90% of users in Twitter and Sina Weibo have fewer than 200 followers and followings, implying that for majority of nodes, the API can be called only once on each visit during the sampling process of USDE.

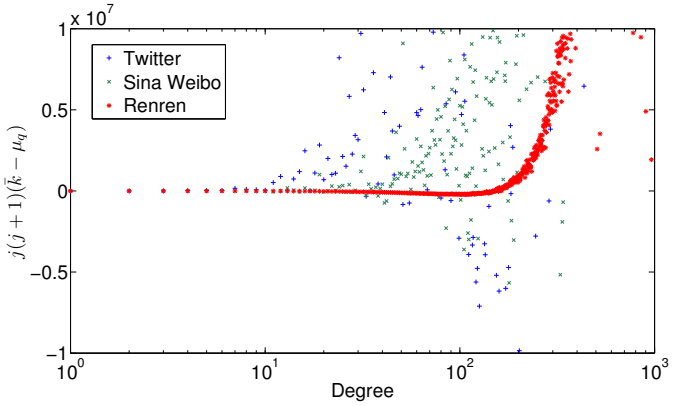


Figure 2-15: The average $j(j+1)(\bar{k}-\mu_q)$ vs. degree in two social media and one online social network

Although the exact local assortativity coefficient ρ cannot be calculated directly (Eq. 2.1) for each node in Twitter and Sina Weibo because of the lack of information on the total number of edges in the whole network, $j(j+1)(\bar{k}-\mu_q)$ in Eq.2.1 can be

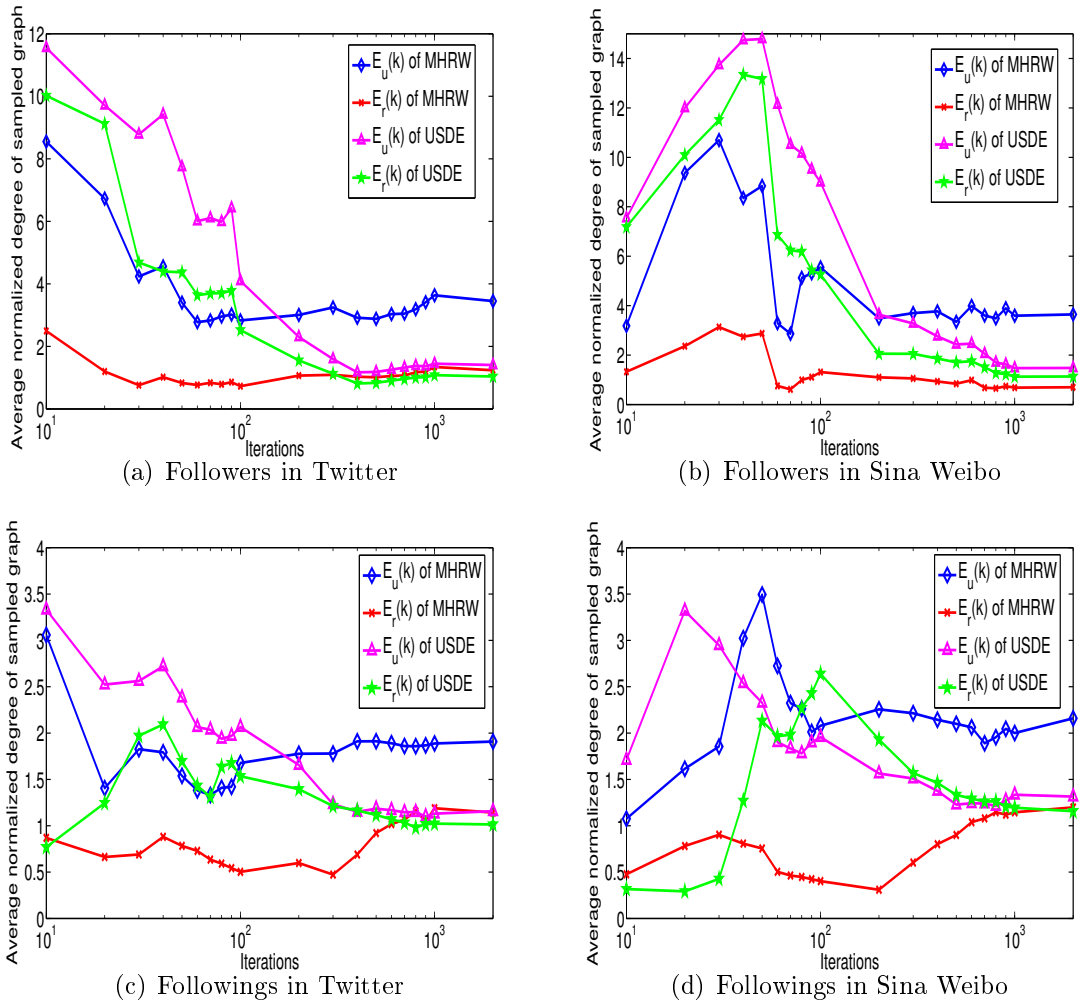
indeed estimated with the help of dataset obtained by the above UNI sampling, where μ_q can be approximated with the average remaining degree of sampled nodes in UNI dataset. Fig. 2-15 shows the average values of $j(j+1)(\bar{k}-\mu_q)$ against degrees in the abstracted undirected networks of Twitter and Sina Weibo. As expected, the local disassortativity is obvious on the nodes with high degree. To illustrate the difference between the online social media and the online social networks, the such values against degrees in Renren network (the largest Chinese online social network) are also ploted based on the dataset provided by [102]. It is expected that the Renren network shows a local assortative mixing pattern regardless of degrees, which is distinct from Twitter and Sina Weibo.

2.6.2 Quality of samples

The work first examines how the sampled average numbers of followers and followings are close to the ground truth in Fig. 2-16. Again, the work compares USDE with MHRW and focus on the values normalized by the ground truth. Large variations of $E_r(k)$ (Eq. 2.6) and $E_u(k)$ (Eq. 2.7) are observed at the initial sampling iterations due to the random choice of seed nodes for the 10 runs. The normalized $E_r(k)$ gradually converges to 1 after taking several hundreds of iterations. Nevertheless, while the normalized $E_u(k)$ in USDE in both social media converges close to 1, $E_u(k)$ for the number of followers (and for the number of followings) in MHRW is about four times (and two times) as high as the ground truth obtained by UNI.

Fig. 2-17 shows the CCDF for followers in Twitter and Sina Weibo obtained by MHRW, USDE and UNI with 2,000 iterations where the sampled users are counted uniquely. It can be observed that USDE generates a closer distribution to UNI than MHRW. For example, in the sampled subgraph of Twitter using USDE, there are 93% users having less than 200 followers, which is close to the corresponding proportion of ground truth 95%. However, the result in MHRW shows only 80% users with less than 200 followers.

The work then compares the two sampling methods from the perspective of accuracy in sampling user attribute. To this end, the work focuses on the user location



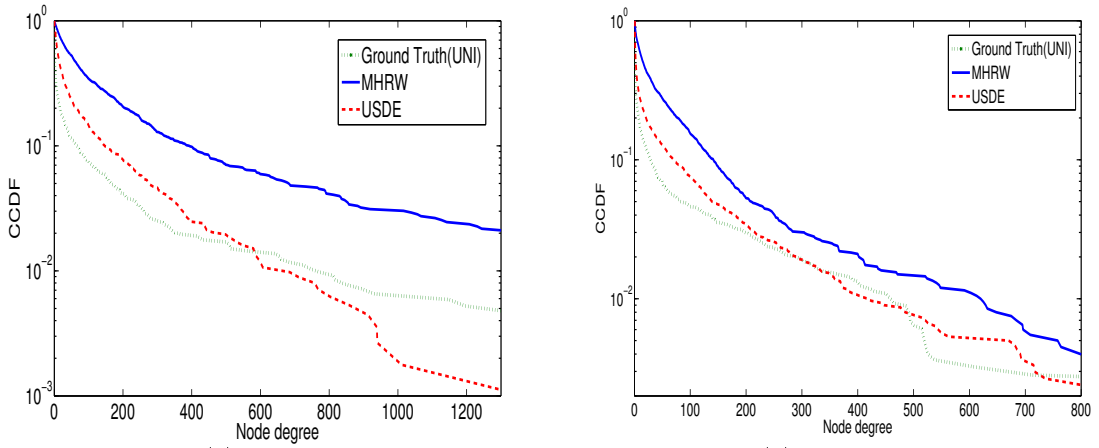
(a) Followers in Twitter (b) Followers in Sina Weibo

(c) Followings in Twitter (d) Followings in Sina Weibo

Figure 2-16: Average normalized number of followers and followings

information. In both Twitter and Sina Weibo, every user has a location profile at the level of city, which is suitable to be used as the standard to measure the performance of USDE and MHRW on user attributes.

Fig. 2-18 makes a comparison between the proportions of locations obtained by two sampling methods in Sina Weibo with 2,000 iterations. The work selects five provinces (aggregated above city-level) to show the difference. It can be observed a limited difference for popular locations (*e.g.* Shanghai, Fujian). However, MHRW misses the locations accounting for small proportions such as Chongqing and Ningxia, which are indeed captured by USDE.



(a) Twitter (b) Sina Weibo

Figure 2-17: Distribution of sample number of followers

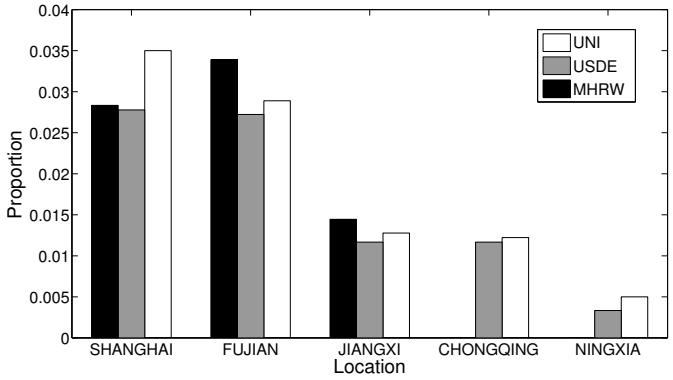
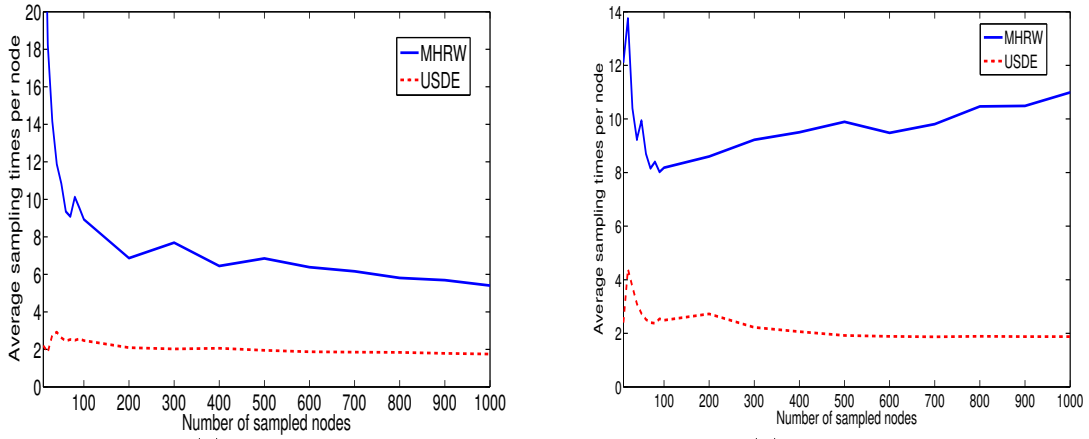


Figure 2-18: The proportions of locations sampled by MHRW, USDE and UNI in Sina Weibo

2.6.3 Sampling Efficiency

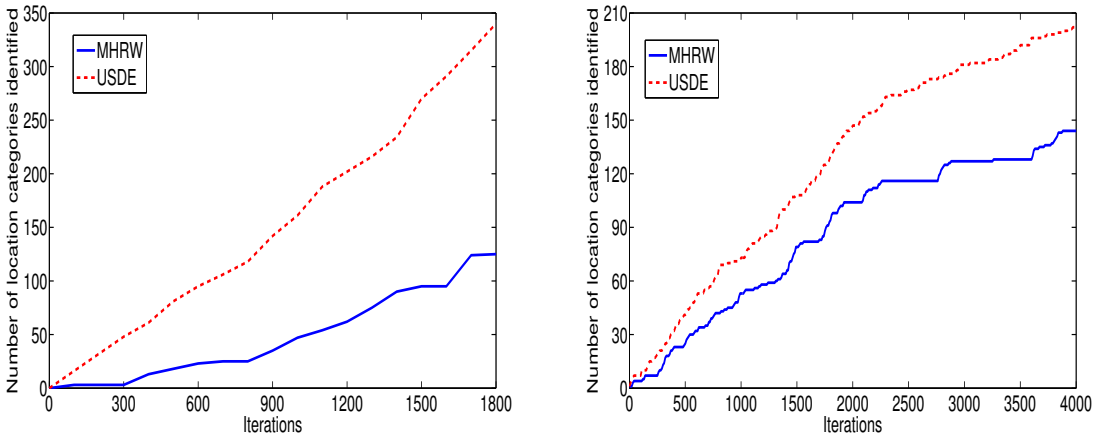
To evaluate the sampling efficiency of USDE, the work first examines the average sampling times per node in Fig. 2-19. For both Twitter and Sina Weibo, the average sampling times per node of USDE are close to 2. In contrast, MHRW has a much higher number of sampling times per node, which is 6-8 and 8-10 for Twitter and Sina Weibo respectively.

Fig. 2-20 then investigates the efficiency in identifying new locations of MHRW and USDE. USDE can consistently identify more new locations than MHRW. For example, USDE identified more than 200 cities in Twitter within 1,200 iterations,



(a) Twitter (b) Sina Weibo

Figure 2-19: Average sampling times per node



(a) Twitter (b) Sina Weibo

Figure 2-20: Efficiency of identifying location information

while MHRW only identified about 50 cities. The difference between the two curves in Twitter is more significant than that in Sina Weibo, possibly due to the categories of attributes in Twitter (world-wide) are more various than the ones in Sina Weibo (country-wide).

2.7 Summary

This chapter studies the unbiased sampling of online social media, which exhibit local disassortative mixing pattern. The work models the random walk-based sampling as

a Markov chain to deduce the general conditions for unbiased sampling, and then proposes a sampling method, called USDE. It introduces dummy edges between nodes with high self-sampling probabilities to allow crawlers to flexibility move between low-degree nodes, while keeping the connectivity of samples. The work has detailed the way of building dummy edges and the computation of moving probabilities, and theoretically analyzed the efficiency of USDE. The performance evaluation results in both synthetic networks and two real-life social media have demonstrated that USDE outperforms previously proposed sampling methods in both the quality of samples and sampling efficiency.

Chapter 3

Information Diffusion in Microblog

3.1 Motivation

Microblog services, such as Twitter and Sina Weibo, have greatly changed the way of information dissemination. The speed and convenience of Microblogs make them competitive services with classical media. With the increase of importance of Microblog as a social medium for information sharing, understanding mechanisms describing how information diffuses over Microblogs, and explaining how some tweets become popular, are meaningful for public opinion mining and information recommendation on Internet currently.

This chapter makes an analysis of tweet's popularity at first. The work invests cascade effect[20] of information propagation in Microblog services where it is assumed that information diffusion proceeds in an eventually random number of successive stages. The aim of this work is to validate the usage of such a cascade effect for describing information diffusion in Microblog services. The analysis unveils that the distribution of tweets' popularity follows the stretched exponential (SE) distribution, instead of the expected power-law distribution, and the parameters of SE distribution can be used to estimated crucial properties of cascade. Moreover, the number of retweets decreases exponentially with the growth of retweeting hop giving preliminary evidence for a simple multiplicative model.

Based on the analysis of cascade effect, Galton-Watson process is introduced to

describe the retweeting process in Microblogs. Galton-Watson process is a traditional multiplicative model which can characterize the cascade effect in the information diffusion in Microblogs well. However, the retweeting probabilities of the tweets always decrease along with the growth of number of hops, leading to the quick stop of retweeting process, while the Galton-Watson process is likely to continue infinitely. To consider the timeliness of the retweeting process in Microblogs, the Galton-Watson with Killing (GWK) process, which is a variant of Galton-Watson process, is proposed in the analysis of information propagation in Microblogs. The work collects the Microblog data from Twitter and Sina Weibo in 2011, and evaluates the performance of the GWK model based on these two datasets. The results of experiments show that GWK model can fit 82% of number of receivers of tweets and 90% of maximum number of hops in the real retweeting process accurately. Besides, the work uses two applications to show that the parameters of GWK model are useful to reveal the endogenous and exogenous factors which affect the popularity of tweets.

3.2 The Multiplicative Cascade Model for Tweet Popularity

A Microblog user might follow another user, *i.e*, he will receive all messages (called *tweets*) sent by the followed person. Followers might retweet some of the messages they receive to their own followers. The distance between retweeters and the tweet's publisher is called *hops*. The retweeting mechanism enables users to spread information to users that could not normally access it. Through retweeting hot messages can be received by tens of thousands of users. The work measures popularity of a tweet by the number of people that have retweeted it.

The retweeting pattern in Microblogs can be described as a random multiplicative cascade process. Formally, a random multiplicative cascade process $X(.)$ can be described at each point k as a multiplication of n random variables $m_1, \cdots, m_n$, *i.e.*, $X(k) = m_1 \times m_2 \times \cdots \times m_n$. To relate this model to the propagation of information,

the work defines that the i-th stage begins at the time for generation of the first retweet at hop i and ends until the first retweet at hop $i+1$ appears, that is, the number of cascade stages is equivalent to the maximum retweeting hop. The number of new users that will retweet a tweet at i-th ($1 < i \leq n$) stage is a coefficient α_i of the overall number of users that have retweeted the tweet up to the $(i-1)$-th stage. Then the overall number of users retweet a particular tweet t after n stages, denoted as $N^n(t)$, is given by $N^n(t) = (1+\alpha_1)(1+\alpha_2)\cdots(1+\alpha_n)$, where the expansion coefficient α_i is in fact related to two main factors: the proportion of followers that will retweet at i-th stage, and the out-degree (*i.e.* the number of followers) of the retweeters.

Similarly to central limit theorem that applies to sum of random variables, an asymptotic limit theorem for multiplicative processes where all multiplied random variables are i.i.d [32] can be derived. Such processes converge to a stretched exponential (SE) distribution defined as Eq. 3.1:

$$P(X \geq x) = e^{-(\frac{x}{x_0})^c} \tag{3.1}$$

where the stretched factor is related to the number of multiplied random variables m, *i.e.* the number of cascade stages, through a simple relation $c = \frac{1}{m}$, and x_0 is a constant parameter that is related to ranking scale. Because of its particular shape a stretched exponential distribution can be easily mistaken with a power law one [59]. However, processes following a stretched exponential distribution will have a particular rank ordering statistic that will be different from the one of a power law. Let i be the rank of an observation from a stretched exponentially distributed process and y_i its observed value. It can be shown theoretically that Eq. 3.2 is valid:

$$y_i^c = -a \log i + b \tag{3.2}$$

where $a = x_0^c$ and $b = y_1^c$, meaning that the modified ranking diagram, showing y_i^c, the observed values with exponent c *vs.* the log of its rank, follows in a straight line with slope $a = x_0^c$. This analysis suggests that if an empirical distribution follows a

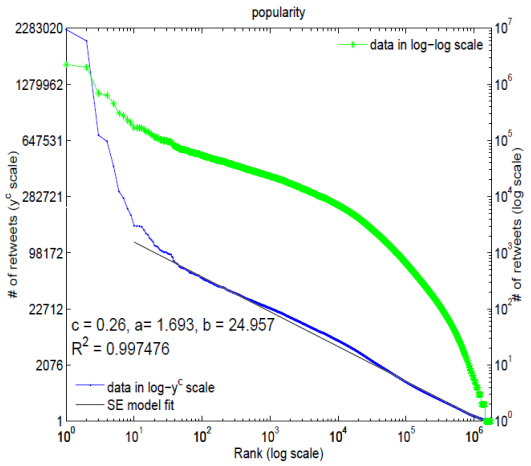


Figure 3-1: Stretched exponential distrution Fitting

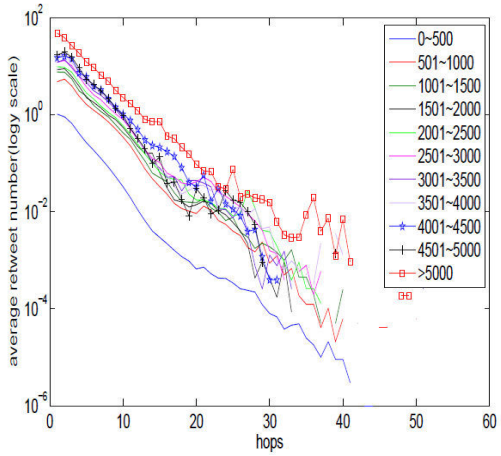


Figure 3-2: Distribution of average retweet number in each hop

stretched exponential it can be meaningful to search for a multiplicative cascade that could explain the emergence of this global distribution. In order to check this, the work fits SE models to the observed tweet popularity rank-ordering distribution using the matlab toolbox provided by authors of [28]. Fig. 3-1 shows the popularity distribution for all collected tweets in both log-log scale and log-y^c scale. The parameters of the SE model along with the R^2 statistic of the fitting are marked in the figure. The figure shows that the SE model fits the distribution very well, except the first several points that are due to the "King effect" [59] (this resulting from the fact that popular topics reduce the attractiveness of other topic because of their high popularity). The work also fits SE for different days and lists in Table 3.1 the obtained parameters showing the relative consistency of the c parameter in close dates.

In particular the SE model predicts that one can expect a number of maximum retweeting hop (a number of cascade stage) around $m = \frac{1}{c}$. Table 3.1 also shows the number of maximum retweeting hop h_e derived using SE model: $h_e = \frac{1}{c}$ and the empirically observed value over the dataset h_a. As can be observed these values are very close. These results give more rational for a multiplicative cascade model of tweet popularity.

To model the multiplicative cascade, the work analyzes how the number of retweets relates to the number of retweeting hops from the tweet generator. For this purpose,

Table 3.1: Parameters of different days of tweets

Time	c	a	R^2	h_e	h_r
06/12/10	0.38	8.617	0.9980	2.63	2.69
07/12/10	0.36	6.523	0.9987	2.78	2.73
08/12/10	0.37	7.979	0.9987	2.70	2.72

the data is stratified into 11 states according to the popularity of the tweets inside it, *i.e.* for $1 \leq i \leq 10$. The i-th state contains the tweets with retweet numbers between $500i$ to $500(i+1)$ and the last set contains all remaining tweets. Fig. 3-2 shows in a semi-log scale the evolution of the average number of retweets as a function of its hop distance from tweet source. Interestingly all curves seems to be almost parallel and to be finely fitted to a straight lines. This indicates that the average retweet number decreases exponentially with the hop distance. This is compatible with a cascade model with a constant value of $E\{\alpha_i\} = \hat{\alpha}$ for all stages. Interestingly these figures mean that the tweet's popularity mainly depends on the retweet number at the first hop (or the two first hops), *i.e.*, the number of followers of the originator that forward the tweet. These observations of cascade effect provide the reasonable evidences of using GWK model to characterize information diffusion in Microblogs in the following section.

3.3 A New Model of Information Diffusion in Microblogs

Analysis of online network topologies and information spreading patterns has laid foundation for explicative models of information diffusion. This section builds an explicative model that takes the network topology of actual information diffusion and characteristics of contents into consideration and describes the process of diffusion comprehensively. This work takes an analogy between the family name evolution and diffusion by retweeting where a family name is carried on only by male descents with offspring and information in Microblogs spreads only by those who choose to retweet it. The Galton-Watson process is a branching stochastic process that has been used

for the evolution and extinction of family names, therefore the Galton-Watson (GW) process is employed to the modeling of information diffusion in Microblog services. However, since information diffusion stops rather quickly online because the novelty of online news wears out with time, while family names die out much slowly, the work includes a killing process in the GW model to take into account such peculiar feature of the online information diffusion. The work collects data from Sina Weibo and Twitter in order to use them in the analysis and evaluation of this Galton-Watson with Killing (GWK) model. The results of experiments demonstrate that the GWK model can describe the pattern of information diffusion in Microblogs very well and can be efficiently used to generate synthetic loads of Microblog online information while still guaranteeing the statistical characteristics in terms of tweets popularity. What's more, the GWK model and its parameters reveal the key features of popular tweets.

3.3.1 Related Work

Here prior work is reviewed on the online social networks and social media; online information diffusion and its analytical models. Most previous work to model information diffusion have considered Independent Information Cascades and Linear Threshold models as building blocks and estimated the properties of the obtained cascades. Different from all the previous work, the Galton-Watson model with Killing process introduced in this work, takes both the topology of Microblog social graph and the intrinsic interest of the message into consideration and therefore can describe the online information diffusion more comprehensively and in an accurate way. This is supported by the validation and comparison between empirical tweets distributions and the synthetic model-based information patterns. Specially, as opposed to previous work, in addition to modeling the diffusion and popularity of online information, this work also presents an asymptotic analysis of the proposed process, which in turn allows the researchers to not only validate the model to fit the actual tweets propagation, but also to use it for tweet load synthesis.

Online social networks and social media

From citation networks to call graphs and group dynamics in newsgroups, human dynamics in a great many forms of interaction has long been studied. The following two have analyzed the topological characteristics of online social networks and online social media which are of particular relevance to this work. Mislove *et al.* analyzed four popular online social networks including Flickr, Livejournal, Orkut, and Youtube and found some basic features about OSNs such as a small world phenomenon and high clustering coefficients [69]. Kwak *et al.* reported on news-media-like characteristics of Twitter [58].

Online information diffusion

These online social services offer a massively amount of data on human interaction and have spurred research on information sharing. Generally speaking, there are two directions for the online information diffusion researches, characteristics descriptions and analytical models.

Cha *et al.* provided an in-depth study of YouTube, including an analysis of popularity evolution [19]. Guo *et al.* analyzed the popularity of various user-generated contents (UGC) and found that the observed rank-ordered popularity distribution is not power-law as expected but is a stretched exponential distribution [44]. Lee *et al.* used a Cox proportional hazard regression model to predict the popularity of online contents [60]. Zaman *et al.* gave a probabilistic collaborative filtering model for predicting the popularity of information in Twitter and found that the most important features for information propagation in Twitter are the identity of the source of tweet and retweeter [83]. Lerman and Gosh conducted an empirical description of news spread process on Digg and Twitter [63]. Ye *et al.* showed how breaking news spread through Twitter and provided metrics for social influence of users [113].Goetz *et al.* used "zero-crossing" approach to research the temporal dynamics of the blogosphere [37]. Gomez-Rodriguez *et al.* developed an efficient approximation algorithm to infer the information diffusion network [38]. Work listed above can be construed

as of descriptive nature and do not answer causality of the phenomena.

Other work focused on building analytical models of popularity and diffusion. Two models are widely used for online information diffusion researches, Independent Cascades(IC) and Linear Threshold (LT) models. Kempe *et al.* introduced in [51] the LT model to find the influential users and maximize the information spread on online social networks. Galuba *et al.* analyzed the diffusion of URLs in Twitter and proposed to use the LT model to predict which users will propagate which URLs [108]. Yang *et al.* developed in [110] a Linear Influence Model (LIM) based on LT models which can predict interactions between nodes in the information dissemination process without requiring the knowledge of the social network graph. On the other hand, IC models were firstly used to analyze the information spread on blogosphere [64][22]. And epidemic model, which is a variation of an IC model was also proposed to make the microscopic characterization of information diffusion process [2][43]. Cha *et al.* introduced the cascade model into the research of information dissemination on online social network such as Flickr [20][21]. Myers *et al.* improved the traditional cascade model in which information can reach a node not only via the links of the social network but also through the influence of external sources [72]. Guille *et al.* developed a concrete model which relies on the IC model and is based on machine learning techniques to predict the temporal dynamics of diffusion in social networks [15]. Similarly, [84] proposed a K-tree based model, established to correct for missing data in information cascades, which makes such a model suitable for all types of cascades.

3.3.2 Dataset Description

This section gives a brief overview of Sina Weibo and Twitter datasets including the collection methodology and their basic properties.

Sina Weibo and Twitter

Twitter is a well-known Microblog service, where a user can unidirectionally follow other users and subscribe to their tweets. Sina Weibo is a Chinese Microblog service

with Twitter-like unidirectional follow relationships. Followers on both services can retweet some of the messages they receive from their followings and these retweets are seen by their own followers. The distance between a retweeter and the tweet's original publisher is measured in *hops* where the publisher is considered at the 0^{th} hop. Retweeting is an easy and popular mechanism to share a tweet with followers.

Sina Weibo makes public two statistics per tweet: the number of retweets and the number of comments for the tweet, both of which represent the popularity of the tweet. However, Twitter does not offer comments to a tweet and thus only the number of retweets is used in this work. In latter sections, the popularity of a tweet is measured by the number of retweets.

Data collection methodology

The study needs the complete set of retweets per tweet. Both Twitter and Sina Weibo provide a search API, to which users input a tweet's identification number (ID) and are returned all retweets of the tweet where the retweeting pathes from the original publisher to retweeters are provided in detail. While for Sina Weibo this API is used, the work uses the methodology from Kwak *et al.* [57] in Twitter to collect followers, followings, tweets, and retweets of *all* Korean Twitter users. This Korean Twitter dataset is referred to as T_{user}.

There are over 100 million users in Sina Weibo as of January 2011, of which size is too big to manage without Sina Weibo company's cooperation. Instead the work uses the unbiased sampling method USDE which is introduced in Chapter 2 to reduce the size of the dataset for this work. With this uniform sampling method an unbiased sample of $500,000$ users is obtained finally.

For each user crawled, the work collects all his tweets which have retweets and are published from Aug. 1st, 2011 to Dec. 1st, 2011. For each tweet sampled, all its retweets are also collected. This user unbiased dataset from Sina Weibo is referred as S_{user}.

Table 3.2: Sina and Twitter dataset summary

Dataset	Time	Tweets forwarded	Retweets	Users
S_{user}	8 $\sim$ 12/2011	261,833	1,996,170	500,000
T_{user}	8/2011	1,133,568	3,316,609	4,332,445

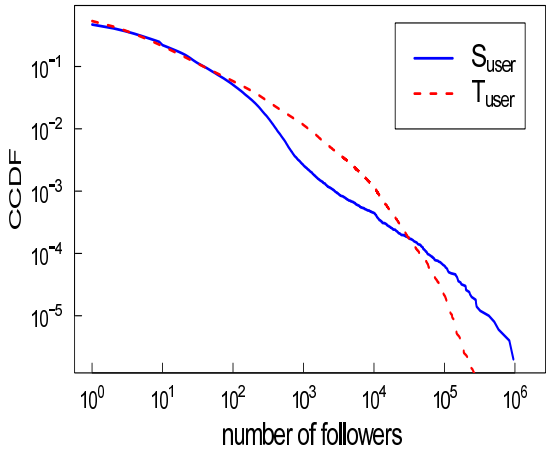


Figure 3-3: CCDF of the number of followers

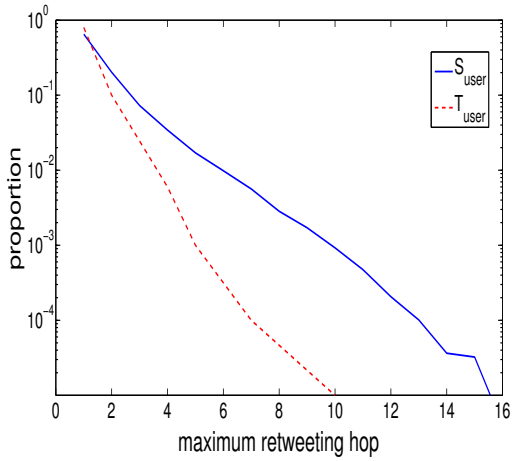


Figure 3-4: Maximum retweet hops distribution in two datasets

Data description

Table 3.2 summarizes the datasets S_{user} and T_{user} characteristics.

Fig. 3-3 plots the CCDF (Complimentary Cumulative Distribution Function) of the users' followers from S_{user} and T_{user}. Neither has a simple power-law distribution. Today's online social networking services often have hundreds of millions of users and are used not only for personal communication but in numerous types of communication, including political campaigns and advertisements. The degree distributions from Cyworld, Twitter, and Orkut have been reported to deviate from a strict power-law distribution [4][58][69].

For each original tweet (message) M, the corresponding retweeting tree $T(M)$ is built as follows. When node S publishes the original tweet M, S is considered as the root of $T(M)$ and all of S's followers which received the tweet are the children nodes of S in $T(M)$. For all children nodes, if node A retweets the tweet M from his parent, then A generates children nodes composed of all his followers, otherwise node A is considered as a leaf node.

Fig. 3-4 plots the distribution of the maximum retweeting hops in the two datasets, S_{user} and T_{user}. This distribution shows the depth of the diffusion and will be used in the forthcoming as a validation metric.

3.3.3 A Galton-Watson Model

The Galton-Watson model has been used with success to model the evolution of family names [78]. The information diffusion process via retweeting online bears a striking similarity to the family name evolution. Family names are transferred patrilineally, while information spreads only by those who retweet. Family generations are analogous to retweeting hops, which indicates the distance between the source of the original tweet and the particular retweet.

One important factor in the above model is the decision to retweet or not. Such a decision depends mainly on the content of the tweet. In family dynamics, it would correspond to fertility. On the other hand, the distribution of number of followers (or descendants) is a topological property of the Microblog social graph (or the family tree). Thus this work takes two important aspects of information diffusion in the retweeting process: the intrinsic interest of the tweet message and the topology of the social graph.

The following will formalize the model at first. A GW is a branching stochastic process $\{X_n\}$, where X_n represents the number of users in the Microblog service that receive a particular tweet through a path of n retweeting hops. The process $\{X_n\}$ is then evolving according to the recurrence formula: $X_0 = 1$ and $X_{n+1} = \sum_{j=1}^{X_n} \xi_j$, where for each generation n, ξ_j is a sequence of Independent and Identically Distributed (IID) discrete random variables following a distribution $f(k)$ representing the number of offspring of a node. For the purpose of the tweet propagation modeling, there is:

$$f(k) = (1 - \alpha)\mathbb{1}_{\{k=0\}} + \alpha\mathcal{D}(k)\mathbb{1}_{\{k>0\}} \tag{3.3}$$

where α is the probability that a user receiving a tweet message can retweet it, and $\mathcal{D}(k)$ is the degree distribution of the Microblog social graph, *i.e.* the distribution of

number of followers for the Microblog.

It is however important to notice that this tweets propagation modeling and the original GW family name process may differ from the process termination's perspective. Typically, a tweet has a shorter lifetime than a family name (in terms of hops) and will inevitably die faster, *i.e.* no more retweeting activity is observed. While a genealogical tree, depending on the distribution of the number of male offspring is more likely to last longer. This can be explained by the fact that online content might be more prone to a platitude effect due to lack of content novelty. In order to account for this peculiar property, this work models an extinction process that will govern the original GW trees.

Analytic approach

First, the GW trees without considering the extinction process is described. The mean evolution of a GW tree can be easily analyzed through the Wald equality that gives $\mathbb{E}\{X_{n+1}\} = \mu\mathbb{E}\{X_n\}$, where $\mu = \mathbb{E}\{\xi\}$ is the mean number of offspring of members, *i.e.* the mean number of people receiving a micromessage retweeted from one user directly in one retweeting process. In other terms, there is:

$$\mathbb{E}\{X_{n+1}\} = \mu^n \tag{3.4}$$

where $n \geq 0$.

Using the previous assumptions expressed in Eq. 3.3, μ can be rewritten as $\mu = \alpha\delta$ where α is the probability of retweeting a message and δ is the mean number of followers of Microblog users that have a least one followers, resulting finally in:

$$\mathbb{E}\{X_{n+1}\} = (\alpha\delta)^n \tag{3.5}$$

where $n \geq 0$.

It should be noticed that for the retweeting process, X_1 doesn't meet Eq. 3.5. When the information source publishes a tweet, all followers of the publisher can receive the message at the first hop which means X_1 is equal to δ but not to $\alpha\delta$. In

this case, Eq. 3.5 should be amended as follows to describe the retweeting process:

$$\mathbb{E}\{X_{n+1}\} = \delta(\alpha\delta)^{n-1} = (\alpha)^{n-1}(\delta)^n \tag{3.6}$$

where $n > 0$ and $X_0 = 1$.

Eq. 3.6 has the interesting property of separating two effects on the information spreading in Microblog services: the intrinsic interest of the message represented by α and the properties of the social graph represented by δ. The mean total number of users receiving a tweet can be derived as $\bar{M} = \sum_{i=1}^{\infty} X_i$ and the mean total number of retweets is derived as $\bar{T} = \sum_{i=1}^{\infty} \alpha X_i$.

A more refined analysis of the evolution of GW trees can be done through the Probability Generating Function (PGF) that is defined for a discrete random variable X with pdf $p(k)$ as:

$$\phi(s) = \mathbb{E}\left\{s^X\right\} = \sum_{k=0}^{\infty} p(k)s^k \tag{3.7}$$

The work defines $\phi_n(s) = \mathbb{E}\left\{s_n^X\right\}$ as the PGF of X_n. In the context of the GW tree it is easy to prove that $\phi_{n+1}(s) = \phi_n(\phi(s))$. Deriving the precise value of the PGF in general is hard and closed form for it is available in a very limited number of cases. However the PGF relationship is useful for deriving asymptotic properties of the GW process. It is easy to see that $\phi_n(0) = p(0)$, which is the probability that the n^{th} generation is the last generation and that $\phi_n'(1) = \mathbb{E}\{X_n\}$, which is the mean number of users receiving the message at n^{th} generation.

There are two cases of interest here. First case is subcritical and happens when, $\delta\alpha < 1$, and the second case is supercritical when $\delta\alpha > 1$. One important parameter is the probability of extinction, *i.e.* the probability that $X_k = 0$ for a k. This probability can be derived as the smallest positive solution q of the equation $s = \phi(s)$, where $\phi(s)$ is the PGF of the number of offspring of a node. The work then derives the parameters for the subcritical and supercritical cases that are observed in practice.

A)Subcritical case ($\mu < 1$): when $m = \phi'(1) < 1$, it can be proved that $q = 1$ is the smallest positive solution. This means that the diffusion tree will surely die. The

number of generations (τ) of a subcritical GW diffusion tree can also be bounded as [100]:

$$\begin{cases} \frac{\log \delta}{|\log \mu|}\left(1-\frac{\log\log \delta - |\log \mu|}{\log \delta}\right)\left(1-\frac{1}{\delta}\right) \leq \mathbb{E}\{\tau\} \\ \mathbb{E}\{\tau\} \leq \frac{\log \delta}{|\log \mu|} + \frac{2-m}{1-m} \end{cases} \tag{3.8}$$

The mean total number of users receiving a tweet (M) and the mean total number of retweets (T) are derived as :

$$\bar{M} = \frac{1}{1-\alpha\delta},\ \bar{T} = \frac{\alpha\delta}{1-\alpha\delta} \tag{3.9}$$

B)Supercritical case ($\mu > 1$): When $m = \phi'(1) > 1$, there is $q < 1$ and the tree will continue to grow with a probability $1-q$. More precisely, the asymptotic behavior can be observed, when $n \to \infty$, and in this supercritical case X_n converges either to ∞ with a probability $1-q$ or to 0 with a probability q. In other words, when $\mu > 1$, one can consider that asymptotically the process will die with probability q at each stage, or express differently which means that $\mathbb{P}\text{rob}\{X_n > 0\} = 1-q$ for n sufficiently large. In the supercritical case, the number of generations can be potentially infinite and results in an infinite number of members of the GW tree. However, in the assumption that the tree will be extinct at some point, the expected numbers of members can be derived as:

$$\bar{M} = \frac{1}{1-\phi'(q)},\ \bar{T} = \frac{\alpha}{1-\phi'(q)} \tag{3.10}$$

where ϕ is the PGF of X_n as defined above.

Killing process

So far, one can already derive from the GW process the probability of extinction and the mean number of generations for a tweet spreading. However as it will be observed in the next section on the empirical data , the dynamic of GW process might capture an overestimated model for the propagation of tweets. In essence, it can be observed that in real life the hop depth of the propagation trees and the mean number of users receiving a tweet are lower than what are predicted by the GW process. As discussed

previously, this difference might be explained intuitively by the difference in nature between the genealogy of offspring which is modeled by the GW process and the actual information spreading that might be influenced by the content novelty.

Therefore to introduce a killing probability π, which represents the probability that the GW process is killed prematurely at the n^{th} generation, is necessary, resulting in a Galton-Watson process with Killing (GWK). The probability that the process is killed after k generations becomes $\pi(1-\pi)^{k-1}$.

One might at the first glance think that a GWK process with retweeting probability α is equivalent to a classical GW process with retweeting probability $\alpha_K = (1-\pi)\alpha$. However, this is not the case because when killing happens in the GWK process, in the last hop where all nodes are stopped together, one cannot assume anymore that nodes have offsprings independently from each others, which is a different assumption compared with the classical GW model. Nonetheless, the GW process with retweeting probability α_K can be considered as a lower bound of the GWK process, with α, the probability of having offspring at any hop (except the last one) in the GW process, being strictly lower than the corresponding probability in the GWK process. For this reason, one can then expect the number of receivers in a GW process with a retweeting probability α_K and $\mu_K = \alpha_K\delta$ to be a strict lower bound of the number of receivers in a GWK process.

In general, in a GWK process, a GW tree falls in one of three situations: either it is finished because of natural extinction of the GW process, or it is killed because of the killing process or it can also grow infinitely. If $\mu < 1$ then the probability of the third situation happening is 0. This typically means that the probability of generating a finite GW tree is larger for the GWK process than for GW. However, a major issue is that one can't disambiguate the reasons why a tree would stop growing because of a natural extinction or a killing process. The next section will show that this might create problems during the estimation of the killing probability π.

Asymptotic analysis of the GWK process

As described earlier, the PGF of X_n in a GW process is obtained recursively from the PGF of the number of offspring $\phi(s)$ through $\phi_{n+1}(s) = \phi_n(\phi(s))$. When a killing probability is added to the GW process the PGF of the overall number of members of the GW tree is given by :

$$\phi_M(s) = \sum_{n=1}^{\infty} \phi_n(s)\pi(1-\pi)^{n-1} \tag{3.11}$$

and it verifies the following equation $\phi_M(s) = \pi\phi(s) + (1-\pi)\phi_M(\phi(s))$. The mean number of members of the GW tree, or equivalently the mean number of users receiving a tweet, can be derived as :

$$\bar{M} = \phi'_M(1) = \sum_{n=1}^{\infty} \phi'_n(1)\pi(1-\pi)^{n-1} \tag{3.12}$$

where $\phi'_n(1) = \mu^n$ for GW process. Therefore this relation for a GW process with a killing probability π can be got as follows:

$$\bar{M} = \begin{cases} \frac{\mu\pi}{1-\mu+\mu\pi}, & \text{when } \mu_K < 1 \\ \infty & \text{when } \mu_K \geq 1 \end{cases} \tag{3.13}$$

In [78] it is showed that if the offspring distribution has finite mean $\mu = \phi'(1)$, the dominant tail will be $P(M = m) \approx R(m)m^{-1-\kappa}$ where

$$\kappa = \frac{\log(1-\pi)^{-1}}{\log \mu} \tag{3.14}$$

This result means that one can expect the distribution of the number of users receiving a tweet to have a power law behavior with an exponent $1 + \kappa$.

3.3.4 Validation

This section will validate the usage of GW and GWK model for describing information diffusion over Microblogs. The work first describes how to estimate the model parameters and thereafter shows that the GWK can be applied to the retweeting trees.

Parameter estimation

The model described in the previous sections depends on three main parameters:α_i the retweeting probability of the tweet i, δ the mean number of followers of Microblog users that have at least one followers, and π, the probability that a tweet diffusion tree is killed at each hop. The work first needs to propose way of extracting these parameters over a dataset. Out of these three parameters the first one has to be derived for each tweet and the last two have to be derived over the whole dataset and can be considered as properties of the Microblog site. A major issue that the work has to deal with is relative to the ambiguity of the cause of death of a diffusion tree. A diffusion tree can be finished because of natural extinction caused by the GW process or of being killed by the killing process. As α is relative to the GW process and π to the killing process, being able to separate these two effects is very important.

A)Inference of δ: For the forthcoming analysis it is necessary to obtain the distribution of the number of followers and to derive from it the mean and its PGF. As explained earlier, the initial user unbiased Sina Weibo dataset contains 500,000 Microblog users. As the number of followers of a user is the open and available information, deriving the follower statistics is straightforward. It is observed on this dataset a mean of 93.4 followers, and a very large deviation equal to 4520.2, showing the very large variability of the number of followers in the dataset. Twitter dataset has a mean of 103.92 followers, and a deviation equal to 1436.8. The resulting distribution is used as $\mathcal{D}(k)$ in Eq. 3.3.

B)Estimation of α_i: The second important parameter to estimate is the retweeting probability α_i. However this value should be estimated when the diffusion tree is not

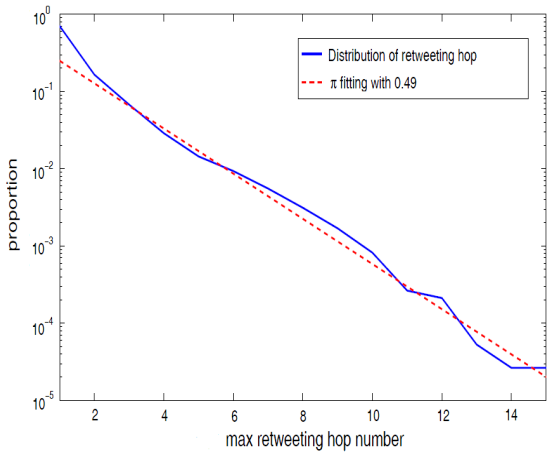


Figure 3-5: π fitting with distribution of maximum retweeting hop number of S_{user}

in the "killed" state. To ensure this, α_i in the retweeting tree of i is estimated as the proportion of users that retweeted the message among all users who received the message excluding these in the last diffusion hop.

C)Estimation of π: The last parameter to infer is π, the probability of killing a tweet at each hop. Therefore it is necessary to ensure that a tweet is not naturally extinct , before using it for estimating π. In order to achieve this, for each diffusion tree the probability that the tree is naturally extinct at the last hop is calculated. If one assumes that a tweet i has a tree with N receiving users in its last hop, and that the retweeting probability of this tweet is α_i, then the probability of extinction is given by $P_e(i) = (1 - \alpha_i)^N$. The work derives for all tweets this value and decides to put aside all tweets that have a probability of extinction larger than 5%, or in other terms, the work focuses on all tweets which the probability of being killed at the last hop is larger than 95%, resulting in 37,866 tweets. Thereafter the value π is derived by fitting the formula $\pi(1-\pi)^l$ to the distribution of maximum retweeting hop number l measured over these tweets. Fig. 3-5 shows the fit over the distribution of maximum hop number of S_{user}. It can be observed that the distribution of generation number follows the expected exponential decrease. The work estimates $\pi = 0.49$ for Sina Weibo and $\pi = 0.53$ for Twitter.

Table 3.3: Galton-Watson parameters in Microblog

Dataset	δ	π	$\%(\mu_i < 1)$
S_{user}	93.4	0.49	54.2%
T_{user}	103.92	0.53	77.5%

GW model validation

As explained in Section 3.3.3, there is a fundamental difference among the two cases: $\mu < 1$ and $\mu > 1$. The GWK process parameters calibrated over S_{user} and T_{user} are shown in Table 3.3. Here one can estimate for each tree a μ_i using the following estimator that is known to be the maximum likelihood estimator of μ:

$$\hat{\mu} = \frac{\sum_{i=2}^{L} X_i}{\sum_{i=1}^{L-1} X_i}, \tag{3.15}$$

where L is the maximum hop length of the retweeting tree. Note that the number of offspring for $X_0(i.e.\ X_1)$ is not considered in numerator, because as stated in Section 3.3.3, Eq. 3.5 is not suitable for X_1.

Table 3.3 also gives the proportion of tweets in $\mu < 1$ case for each dataset. The work has first to assess if the killing process is needed or not. For this purpose the work can do two tests: first to check if the number of observed hops is compatible with the formula given in Eq. 3.8, and to check if the mean number of receiving nodes is compatible with Eqs. 3.9 and 3.10. The first method is only applicable to subcritical tweets and the second method is applicable to all tweets. Using the lower bound in Eq. 3.8, it can be observed that 89% of tweets have a maximum retweeting hop number less than the lower bound given for the GW model. Moreover, Fig. 3-6 shows the number of receivers as predicted by the Galton-Watson model and what is observed. It can be observed that for 83% of tweets, the observed receivers number is less than the value predicted by the Galton-Watson model (below the line $y = x$ line). Indeed, one can expect that tweets which are naturally extinct will have a mean receivers number following the GW equations. However it can be observed that there is considerable proportion of tweets that have a number of receivers much less than the mean predicted by the GW model. The two above results validate that it

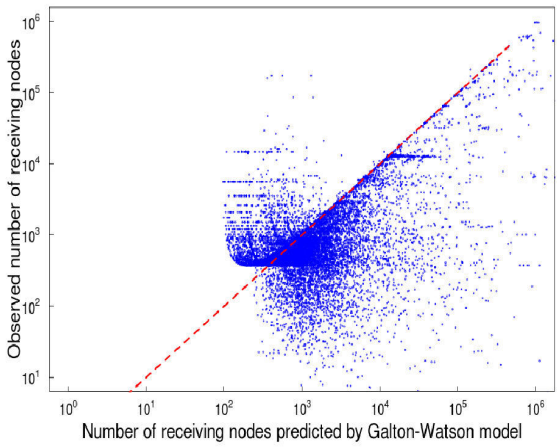


Figure 3-6: Comparison of number of receivers in a tweet tree as predicted by the GW model with what observed.

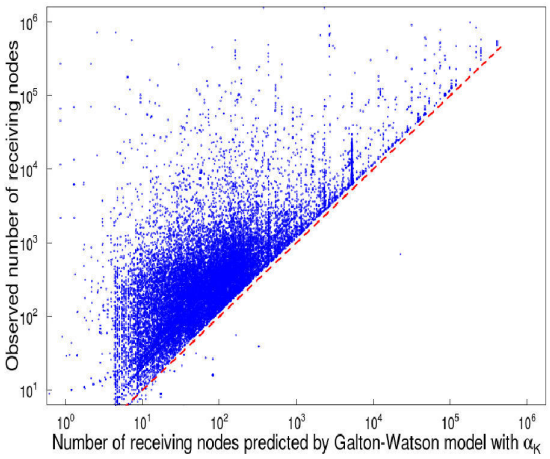


Figure 3-7: Comparison of number of receivers in a tweet tree as predicted by the modified Galton Watson model with what observed.

is necessary to add a killing process that will account for the reduction of number of receivers and the smaller hop lengths.

Section 3.3.3 explained that the number of receivers in a GWK process can be lower bounded by the number of receivers in a modified GW process with retweeting probability α_K. Fig. 3-7 shows the comparison of the number of receivers predicted over the modified GW process with what is observed. The figure confirms that the modified GW process acts as a strict lower bound to the GWK process. However as expected this bound is not very tight.

The above analysis validates the relevance of the GWK process for analyzing the propagation in Microblog systems. In the following the work presents two possible applications demonstrating the usefulness of the GWK model.

3.3.5 Applications

This section presents two applications of the GWK model. The first application shows the use of the proposed GWK model in synthetic workload generation with similar statistical properties to empirical Microblogs load, validating that the proposed model is constructive. This opens way to implement Microblog traffic simulators that can be used to stress Microblog systems. The second application is relative to highly

popular tweets. The GWK model and its parameters provide fine grain features that will be shown to be highly relevant to understanding the popularity of tweets.

Tweet load synthesis

The GWK model provides a way of generating tweet propagation trees by simulating the GWK model with parameters derived over an empirical dataset. The simulation can be easily implemented as it simply consists of beginning from one seed user and generating the first generation by choosing a number of receivers following the distribution $f(k)$ defined in Eq. 3.3. Recursively, each user of a new generation chooses its receivers number following the same distribution. At the end of each generation one checks with probability π if the generation should be killed. The parameter π, and the distribution $\mathcal{D}(k)$ are obtained following the above described methods. The parameter α_i is chosen randomly from an empirical distribution obtained over the corresponding dataset. This can be implemented in a small program that generates trees similar to the ones generated by Microblogs. Fig. 3-8 shows the Complementary Cumulative Distribution Function (CCDF) of receivers number in the trees generated following the GWK model and compares it with the empirical CCDF obtained over the dataset S_{user}. As can be seen there is a very good fit between the two distributions both in the head and the tail. Fig. 3-9 also shows the distributions of the maximum retweeting hop number both in the tree synthesized by the GWK model and what observed empirically over the dataset S_{user}. Here the fit is also striking.

The above analysis shows that the GWK can be used to synthesize retweeting trees that have realistic macroscopic distribution. This validates the use of the GWK model for Microblog workload simulation.

Popular tweets characteristics

The asymptotic analysis of the GWK shows the importance of the $\kappa = \frac{\log(1-\pi)^{-1}}{\log \mu}$ in predicting the tail behavior of popularity, measured by the number of retweets, and the audience, measured by number of receivers. This feature is interesting as it mixes in a single equation all the parameters of the GWK and weights the impact of each

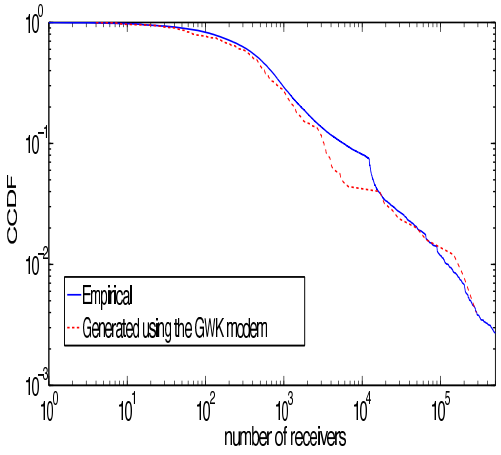


Figure 3-8: Comparison of the CCDF of receivers in trees generated by the GWK model and the empirical CCDF over S_{user}

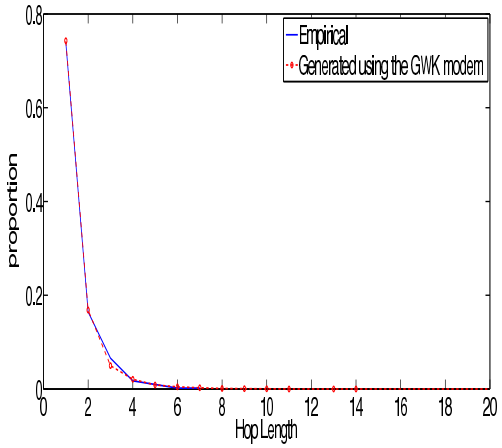


Figure 3-9: Comparison of distribution of maximum retweeting hops in trees generated by the GWK model and the empirical distribution of maximum retweeting hops in S_{user}

of these parameters. It is therefore interesting to look at the value of this feature for overall tweets. However a more precise look shows that as π is a parameter of the global Microblog and is constant for all tweets, so that κ is directly related to $\mu = \alpha\delta$.

Previous works [89] showed that there are two paths for a tweet to become popular: a path that is endogeneous and involves having retweeters that have a large number of followers so that the tweet attains a large audience, and an exogeneous path that explains the popularity by the intrinsic interest of the message. Each one of these paths can be represented in $\mu = \alpha\delta$ where δ accounts for number of followers and α accounts for the intrinsic interest of the tweet, meaning that μ mixes these two aspects. The theorical analysis shows a clear distinction of the asymptotic behaviors between $\mu > 1$, where the tree is expected to become infinite in absence of killing, and the case $\mu < 1$ where the tree will surely extinct even without killing. It will be interesting to check if tweet audience and popularity, measured in term of number of receivers and number of retweets, are related to the value of μ.

Figs 3-10 and 3-11 plot the relation between the estimated μ using Eq.3.15 for each retweeting tree and the popularity evaluated as the number of retweets. While one would expect that large μ leads to large popularity, this is definitely not the case.

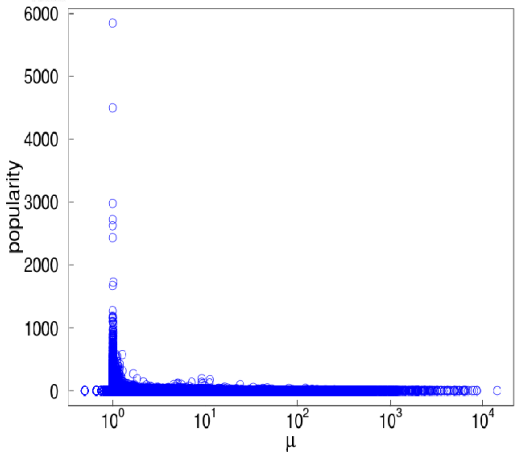


Figure 3-10: Popularity against estimated μ in S_{user}

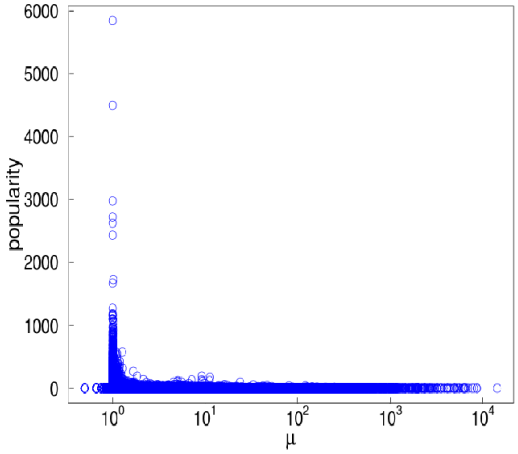


Figure 3-11: Popularity against estimated μ in T_{user}

Table 3.4 contains the characteristics of highly popular tweets. As can be seen all highly popular tweets in the two datasets are sharply concentrated around a value of $\mu = 1$. In fact it can be observed that what leads to a large diffusion and a large popularity is rather a balance between followers number and retweeting probability that results in μ being close to one. This is confirmed by looking at the CCDF of the estimated specific δ_i for each retweeting tree instead of the global constant δ which is shown in Fig. 3-12. Here δ_i is got from $\delta_i = \frac{\mu_i}{\alpha_i}$ where the $0th$ hop and $1st$ hop are not considered as Section 3.3.3 stated. As can be seen in S_{user} the popular tweets generally exhibit δ values that are generally larger than those observed over the whole dataset, but still these tweets have not very large δ. The situation is slightly different in T_{user} where all popular tweets have smaller δ's than other tweets. Nonetheless, either for Sina Weibo or Twitter dataset, popular tweets never happen for small δ's.

The above observations give an interesting characterization of highly popular tweets. These tweets have a δ relatively large (larger than 50 for Twitter and larger than 200 in Sina Weibo) and accordingly small retweeting probability (to end up with a μ close to 1). In particular no popular tweets resulting from several hops of small neighborhood diffusion has been observed, ruling out social rumors type of propagation. It is also observed that few popular tweets have a large value of δ excluding the followers of publisher. This last observation is in accordance with [18].

Table 3.4: Characterization of highly popular tweets

Dataset	Popularity range	Popularity%	Mean of μ	Median of μ	1-percentile	99-percentile	Tweet%
T_{user}	All popularity	100%	3.691	1.247	1.000253	28.5956	99%
T_{user}	popularity $>$ 100	0.182%	1.053	1.002	1.000011	1.271943	51.13%
T_{user}	popularity $>$ 1000	0.002%	1.003	1.000	1.000006	1.028369	17.83%
S_{user}	All popularity	100%	3.596	1.000	0.984	34.1492	99%
S_{user}	popularity $>$ 100	3.06%	1.394	0.8583	0.7676	1.0262	66.8%
S_{user}	popularity $>$ 2000	0.09%	0.904	0.9376	0.8237	1.002	1.5%

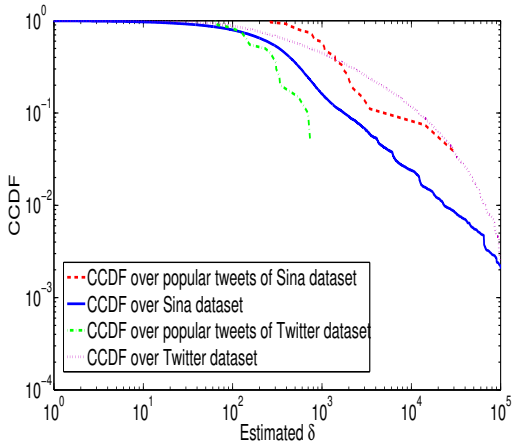


Figure 3-12: CCDFs of δ estimated from popular and all tweets

3.4 Summary

This work has analyzed information spreading patterns in Microblogs and built a novel model for the information spreading. The model is based on the analogy with the Galton-Watson branching processes that describe the evolution of family names. The work has refined the model with the killing process and validated the applicability over two datasets from Sina Weibo and Twitter. Besides, the work has presented two applications of the GWK model, namely, Microblog workload generation and popular tweet characterization, showing that the GWK model is useful not only for describing the information diffusion but also for providing the insights into popular tweets.

This work is leading researchers to the following new directions. The GWK model incorporates time as a discrete generational index, and does not account for the temporal dynamics of tweet diffusion. A continuous time model that captures the temporal dynamics should be considered in the future. In GWK model the values of the mean number of followers, δ, the retweeting probability, α, and the killing

probability, π are used, but the factors behind them have not been addressed. The user's social capital in the network as well as the history of retweets are likely to be correlated to the parameters of GWK model. In addition one particular observation of interest is that highly popular tweets all have a μ value close to 1. The study of compounding factors is left for future work.

Chapter 4

Correlation Analysis between Microblog Trends and Web Interests

4.1 Motivation

The social nature of the Web 2.0 age leads to new patterns of web access and Internet resources discovery. Several studies addressed the interaction between online social media and the popularity of online digital content [31][106][27]. However, the relationship that might exist between online information diffusion in Microblogs and web interest has been overlooked.

While the popularity of contents reflects the long term importance of some content from the Internet users' perspective, trends and in particular positive trends express arising and short-term interests. This work studies the notion of trendiness in the Microblog environment, as one instantiation of online social networks, and in the web context. In the following, the two environments are referred as Spheres. The work focuses on highly positive trends and their corresponding trending topics which attract relatively higher interests within a short period of time.

While there is no absolute metric that captures the interests within the web spheres, this work defines web interest as the extent to which web resources (*i.e.* webpages) are being used or searched in the Internet. Specifically, the work measures interests in the web sphere as the relative number of users who search for a particular

web content using a set of keywords through search engines (*e.g.* Google). In addition, web interest is also measured by the audience of webpages relying on statistics provided by Alexa[5]. Similarly, the work uses established information theory metrics to extract topics of interest and measure trends in the Microblog sphere based on a dataset of tweets collected from Twitter. Furthermore, the work uses official trend statistics from Twitter[99].

This work provides the following contributions: First, by considering a dataset of tweets extracted from Twitter and statistics extracted from Google Trends, the work examines the temporal evolution of trendiness in Twitter and their interrelation with web trends. The work measures the likelihood that a Twitter trending topic is also a web trending topic (as illustrated by Google searches), and characterizes the temporal offset between trendiness in both spheres.

The work finds evidences that trending topics are similar within the two spheres. The results of experiments suggest that trendiness seems to be in most cases originating from the Microblog sphere, with more than 65% of the topics trending in Microblog first. On the other hand, the work also observes that more than 60% of the trending Twitter topics are likely to be also trending in the web and more than 72% of the web trending topics have been (or will be) also trending in the Twitter sphere. A notable difference is that trendiness in Twitter is highly unstable, with almost all Twitter trending topics exhibiting a very low rank stability, as opposed to a high stability observed from the web sphere.

Besides, the analysis is extended to a spatial measurement of trendiness by observing trending topics across five different countries. The work finds both in Twitter and in web, the majority of trending topics appear at not more than 2 countries at the same time (95.6% in Twitter and 65.0% in web) and for a topic, the trending regions in the two spheres are similar, which advocates for a regional feature of trends.

Finally, the work shows theses observations can be used for a "smart" predictive choice of ad keywords in Search Engine Marketing (SEM).

4.2 Related Work

This section first reviews prior work related to the study of the interaction between information spreading on online social media and content popularity.

Some studies focused on the temporal analysis, *i.e.* the co-occurrence in close time interval of popularity growth and the diffusion of information on online social media. Sadikov *et al.* in [31] extracted from online blogs and comments, a comprehensive set of movies features that are thereafter used to predict the corresponding movies sales. Authors in [106] studied the correlation between the popularity of videos on a User Generated Content website (*e.g.* YouTube or YouKu) and the spread of the video urls by tweets. In [47], Teevan *et al.* compared "simultaneous" search queries over Microblogs platforms and on search engines and observed that Twitter searches are mainly used to follow up an event while web searches are mainly intended to learn about a topic [47].

Other studies targeted the spatial dimension, *i.e.* the relationship between the location where a message is published and the scope of its diffusion. Brodersen *et al.* found that social sharing generally widens the geographic reach of a video content. However, when a video cannot generate a social impulse to broaden its paths of discovery, it frequently gets caught in a confined geographic region [16]. Scellato *et al.* described how geographic information extracted from social cascades on online social networks can be exploited to improve caching of multimedia files in Content Delivery Networks [85].

As opposed to previous work, this study provides the first comparative analysis of the rise of interests in the Internet through the comparison of topic trends in Microblogs and on search engines from temporal and spatial perspectives.

4.3 Methodology and Dataset Description

This section first describes the methodology used to infer the trending topics from tweets in Twitter as well as in Google and Alexa, two popular sites that provide trends

in the web sphere. It also introduces the metrics used to measure the trendiness of topics. Finally the datasets used for the analysis is detailed.

4.3.1 Identifying Trends

Trends describe the popularity dynamics of topics over a short time period, where a topic c consists of a word or a sentence mentioned in tweets or queried using search engines. While a single-word topic might be easy to obtain from tweets or queries, multi-words topics should be learnt using some natural language processing methods, *e.g.* LDA.

Trending index volume

User interests in both Microblog and web spheres have temporal dynamics. That is to say, the volume of mentions or searches for a particular topic naturally varies over time. The work defines the *trending index volume* $V_i(c)$ for a topic c at a given time i as the volume of the topic normalized by the maximum volume observed during an observation period of time and then scales the trending index volume to $[0, 100]$, which is similar to the official definition provided by Google[40]. Over a given period R, all trending index volumes $V_i(c)$ where $i \in R$ compose the trends of topic c during that period, $V(c)$, *i.e.* $V(c) = \{V_i(c), i \in R\}$. The work further uses $V^{\mathcal{G}}(c)$ and $V^{\mathcal{T}}(c)$ to represent the trends of topic c in Google and in Twitter respectively.

Extracting the trending index in Twitter is a challenging task, as it is necessary to extract the *global* trending topics over a particular period of time. Although Twitter offers an official trending service, the trends are determined by an "algorithm tailored for the user based on who [you] follow and [your] location. This algorithm identifies topics that are immediately popular, rather than topics that have been popular for a while or on a daily basis, to help [you] discover the hottest emerging topics of discussion in Twitter that matter most to [you]" [97]. In other words, Twitter official trending topics are personalized to user accounts. The work therefore adopts an alternative approach to extract global Twitter trends.

A topic consists of a single word or multiple words. For a single-word topic that includes only one word w, the work measures the trending index as the word frequency based on the content of tweets. The work bins all tweets into subsets S_i with a fixed time interval (daily and hourly in this study) and then extracts, for each subset S_i, the set of words W_i, and computes the word frequency $TF_i(w)$ for each word $w \in W_i$. Note that stop words (*e.g.* "a", "after", "that", *etc.*) which naturally appear with higher frequencies should be ignored in the calculation(readers could refer to [41] for a complete list of the stop words). A word is counted once per tweet even if it is repeated in the tweet. Since the number of tweets in each subset might vary greatly, the word frequency $TF_i(w)$ is normalized by the number of tweets in each subset, resulting in a relative topic frequency $RF_i(c) = \frac{TF_i(w)}{|S_i|}$, where $|S_i|$ is the number of tweets in subset S_i. Finally, the work scales all $RF_i(c)$ in $[0, 100]$. The Twitter trending index volume for single-word topic c at time i in a period R can then be written as: $V_i(c) = \frac{RF_i(c)}{max_{j \in R}(\{RF_j(c)\})} * 100$.

To obtain multi-words topics in Twitter, Latent Dirichlet Allocation (LDA) is used [12]. LDA is a generative model that extracts statistical properties of text documents in a discrete dataset and models each document as a mixture of various latent topics. A topic created by LDA is always nameless and represents a cluster of words that tend to co-occur with a high probability within the topic. LDA learns the statistical relations among words and documents and then estimates the probability that a given document is related to a given topic. The total number of topics is denoted by k, a parameter of the LDA model. Supposing there are M documents in the copus and each document i includes N_i words, the topic distribution θ_i for each document i is described to follow a Dirichlet distribution $\mathcal{D}(\vec{\alpha})$, where $\vec{\alpha}$ is a parameter vector of the Dirichlet prior with a size of k. In addition, the word distribution ϕ_z for a topic z also follows a Dirichlet distribution $\mathcal{D}(\vec{\beta})$, where $\vec{\beta}$ is another parameter vector of the Dirichlet prior. Given the parameters $\vec{\alpha}$ and $\vec{\beta}$, the generative process for each document by LDA contains the following three steps:

1. Choose the topic distribution for a document θ_i from $\mathcal{D}(\vec{\alpha})$, where $i \in \{1, \ldots, M\}$;

2. Choose the word distribution for a topic ϕ_z from $\mathcal{D}(\vec{\beta})$, where $z \in \{1, \ldots, k\}$;

3. For each of word position j in document i, where $j \in \{1, \ldots, N_i\}$ and $i \in \{1, \ldots, M\}$:

   a) Choose a topic $z_{i,j}$ from $\mathcal{M}(\theta_i)$ where $\mathcal{M}(\theta_i)$ is a categorical random variable with parameter θ_i.

   b) Choose a word $w_{i,j}$ from $\mathcal{M}(\phi_{z_{i,j}})$ where $\mathcal{M}(\phi_{z_{i,j}})$ is a categorical random variable with parameter $\phi_{z_{i,j}}$.

Following the above process, the total probability of the model is:

$$P(\vec{W}, \vec{Z}, \vec{\theta}, \vec{\phi} | \vec{\alpha}, \vec{\beta}) = \prod_{i=1}^{M} P(\theta_i | \vec{\alpha}) \prod_{j=1}^{k} P(\phi_j | \vec{\beta}) \prod_{t=1}^{N} P(z_{i,t} | \theta_i) P(w_{i,t} | \phi_{z_{i,t}}) \tag{4.1}$$

where $\vec{W}$ is the set of words in all documents, $\vec{Z}$ is the set of topics in all documents, $\vec{\theta}$ is the distribution vector with size M of which the item θ_i represents the topic distribution in document i, and $\vec{\phi}$ is the distribution vector with size k of which the item ϕ_z represents the word distribution in topic z, N represents total number of words in all documents, that is, $N = \sum_{i=1}^{M} N_i$. The observable variable is $\vec{W}$ while $\vec{\alpha}$, $\vec{\beta}$, $\vec{\theta}$ and $\vec{\phi}$ are latent variables. Note that Eq. 4.1 describes a parametric empirical Bayes model and one can derive various distributions (*e.g.* the associated word probabilities in a topic, the probability that a document belongs to a topic) using Bayesian inference. Gibbs sampling is widely-used to recover the posterior marginal distribution of $\vec{\theta}$.

In the context of this work, all tweets are binned into subsets S_i with a fixed time interval (hourly or daily). In the training process, each tweet is considered as a document and each subset S_i as a corpus of documents. For each corpus, LDA is used with 2,000 iterations of Gibbs sampling to extract 50 topics (*i.e.* $k = 50$), each of which includes 20 relative words. For each training process over S_i, LDA model provides a probability vector for each tweet, the elements of which indicate

the correlation between the tweet and the extracted topics. Based on this probability vector, a tweet can be considered to be related to the topic of which the corresponding probability is the highest in the vector, resulting in the relative topic frequency $RF_i(c)$ (*i.e.* the proportion of tweets related to c in S_i). Then, the Twitter trending index volume $V_i(c)$ for a multi-words topic c at time i can be calculated by scaling $RF_i(c)$ within $[0, 100]$.

The trending index volumes for topics in Google is much easier to be obtained, as Google Trends provides the normalized search volume for both single-word and multi-words topics. These statistics can be used for the computation of the trending index volumes in Google directly.

However, it is hard to get the exact search volumes of topics from Alexa. The work alternatively estimates the trendiness of topics in Alexa approximately with the assistance of topic rank information: the trend of topic c is considered in binary, that is, if topic c appears in the top trending list of Alexa at time i, then the trending index volume of c at i is 100, otherwise, it is 0. Clearly, a sharp rise can happen on Alexa at time i if c is in the top trending list at time i but not at time $(i-1)$.

Positive and negative trends

A topic c experiences a positive (*resp.* negative) trend at time i if its trending index value $V_i(c)$ is larger (*resp.* smaller) than $V_{i-1}(c)$. The corresponding increasing (*resp.* decreasing) trending index volume $V_i^+(c)$ (*resp.* $V_i^-(c)$) is $V_i(c) - V_{i-1}(c)$ (*resp.* $V_{i-1}(c) - V_i(c)$). For a topic c, the work defines *highly positive trend* as a positive trend that has an increasing trending index volume larger than a threshold α at the time of observation. The time of observation i is called *highly positive time (day or hour)* of the topic.

In this study, α is set to be the 50^{th} percentile, 75^{th} percentile and 90^{th} percentile of all positive trending index volumes in $V(c)$.

Trending topics

Trending topics are topics of which trending index volumes increase in a relatively higher proportion compared to others. In other words, a trending topic can be either a word, an expression (a set of concatenated words) or a tweet of which the immediate popularity is rapidly increasing, compared to other popular topics. The emergence of trending topics is either endogenously driven by a users interests, or motivated by an exogenous event that prompts people's attention.

In detail, the work identifies a trending topic at time i as follows:

1. The work derives a discrete-time vector of trending index volumes for each topic c, from which all positive trends are extracted.

2. For each positive trend (of all topics), the work measures the corresponding increasing trending index V_i^+ and then calculates the average value of all increasing trending index volumes at time i, $\bar{V}_i^+$.

3. If at a particular time i, a positive trend of topic c is observed, $V_i^+(c) \geq \bar{V}_i^+$, then the topic c is deemed trending at time i. Time i is called *trending time (day or hour)* of the topic c.

Again, it is noteworthy that the notion of "trending" is different from "popular". The latter is highly dependent on the number of times the topic is mentioned, *e.g.* the number of relative tweets in Twitter or search volume in Google, across a rather long period of time, while trendiness focuses on the speed of increase in mentioning a topic within a short period of time. A topic that has been popular for a while is most likely to be not trending anymore, as the number of tweets mentioning this topic would become steady even though still high.

4.3.2 Datasets

For the purpose of this study, Twitter's tweets are used to extract the trends of topics in Microblog. The work also relies on the "official" trending topics as shown by the

Twitter for geographical pattern analysis. Google Trends and the Alexa services are also used to obtain trends of the web sphere.

Twitter Tweets

The work uses a set of tweets $\mathcal{T}$ from [111] comprising 132,210,436 tweets published by 7,404,248 users over the period from August 1st, 2009 to August 31st, 2009. Two time granularities are considered: a daily topic analysis which matches the Google Trends service [40] time granularity, and a topic extraction on an hourly basis which matches the Alexa trends analysis. As in [116], the work observes that the frequency of the top 5% popular words account for more than 95% of words count in the overall daily and hourly subsets of tweets $\mathcal{T}$.

The work extracts daily (*resp.* hourly) single-word topics using simple term frequency statistics to extract the most relevant (top 5%) words on a daily (*resp.* hourly) basis. To extract multi-words topics, the LDA generative model is used as described above to classify them into different topics. In total, the daily set of topics, denoted as $K_d^{\mathcal{T}}$ is composed of 76,760 single-word topics and 267 multi-words topics, while the hourly set of topics $K_h^{\mathcal{T}}$ is composed of 56,774 single-word topics and 372 multi-words topics.

Official Twitter trending topics

The original tweets collected do not provide enough geographic information. In order to analyze the geographic patterns of Twitter trending topics, the work further collects for the period spanning from September 1st to October 31st, 2012, and every five minutes, the top 10 trending topics suggested by Twitter for the following countries: U.S, U.K, Canada, France and Australia, which are abbreviated to US, UK, CA, FR and AU later. Finally, 6,858 unique trending Twitter topics are found, which compose a topic set H.

Google Trends

For the purpose of temporal analysis, this work collects the Google Trends statistics of the topics extracted from Twitter. Though Google Trends, a dataset, referred as $\mathcal{G}$, can be got which includes scaled and normalized daily Google search volumes for each topic $c \in K_d^{\mathcal{T}}$ from August 1st to August 31st, 2009, leading to the dataset $K_d^{\mathcal{G}}$. In addition, in order to have a comparison study of geographical patterns in Twitter and Google Search, the dataset $\mathcal{G}$ also includes the lists of top 10 countries where the topic $c \in H$ is the most frequently searched topic according to Google Trends from September 1st to October 31st, 2012, leading to the dataset $H^{\mathcal{G}}$.

Alexa rank lists

Although Google Trends provides the trending topics in Google [40], it is not suitable for the web trending topics collection mainly for two reasons. First, Google Trends service only offers the top 10 trending topics per day which are far from enough to compose a complete trending topics set. Second, Google Trends provides the daily trending topics of specific countries but not the global ones. Fortunately, some informative websites such as Alexa provide the information about global trending topics in the web with a fine granularity. Specifically, Alexa keeps track of the top 20 global trending topics (search keywords) in the web for any hour since July 26th, 2009. This provides an effective way to estimate the hourly trendiness of topics in the web sphere. The work collects the hourly top topic lists of Alexa during August 1st to August 31st, 2009. Totally, there are 898 unique trending topics, composing the web topic set $K_h^{\mathcal{A}}$. This dataset includes information about the topics' ranks in each hour as well.

Table 4.1 summarizes the datasets used later.

4.4 Temporal analysis

This section investigates how the trending topics in Microblog sphere behave in the web at first. Later, it proceeds to analyze the reverse interrelation by studying the

Table 4.1: The summary of datasets

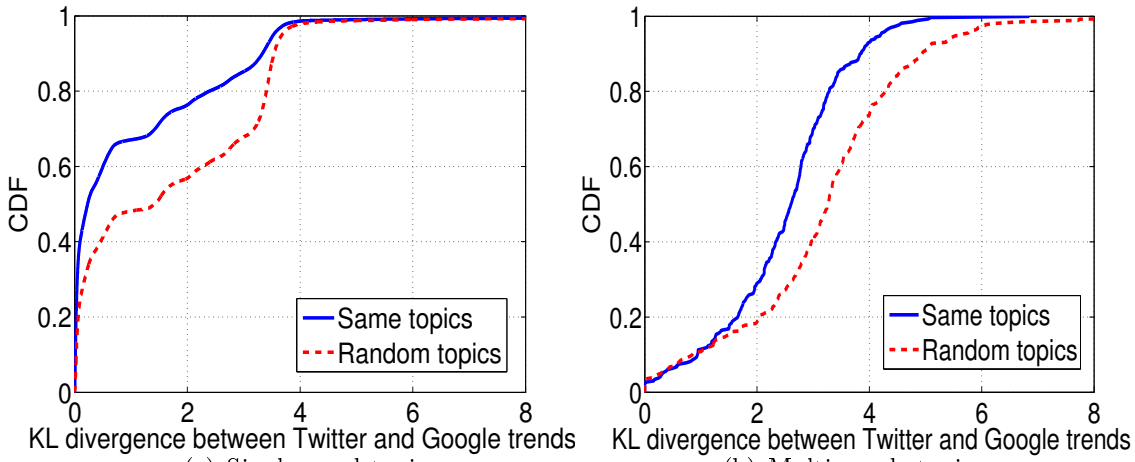
Dataset	Description
$\mathcal{T}$	Dataset collected from Twitter
$\mathcal{G}$	Dataset collected from Google
$\mathcal{A}$	Dataset collected from Alexa
$K_d^{\mathcal{T}}$	Twitter topics set extracted from $\mathcal{T}$ with daily granularity
$K_h^{\mathcal{T}}$	Twitter topics set extracted from $\mathcal{T}$ with hourly granularity
$K_d^{\mathcal{G}}$	Google topics set extracted from $\mathcal{G}$ with daily granularity
$K_h^{\mathcal{A}}$	Alexa trending topics set extracted from $\mathcal{A}$ with hourly granularity
H	Dataset including the official Twitter trending topics of 5 different countries
$H^{\mathcal{T}}$	Dataset including the official Twitter trending topics of 5 different countries
$H^{\mathcal{G}}$	Dataset including the official Google trending country lists of topics in $H^{\mathcal{T}}$

Alexa dataset compared to the collected Twitter dataset to examine how the trends of trending topics in the web look like in Twitter.

In summary, it is found that trending topics are similar within the two worlds where at least 70% of the Twitter trending topics are likely to be also trending in the web and 72% of the web trending topics have been (or will be) also trending in the Twitter world. The results also suggest that although the trendiness in Twitter seems to be synchronous with the one in Google on daily granularity basis, most of the trends of these topics are actually driven by Twitter population in advance, and then spread in the web on a finer granularity (such as on an hourly basis). The notable observed difference is that trendiness in Twitter is highly unstable. It is found that almost all Twitter trending topics exhibit a very low rank stability, which is opposed to the high stability observed for the web trending topics.

4.4.1 How do Twitter topics behave in the web?

As the topics extracted from tweets are used to collect their trends in Google, the work can have an analysis on how accurately topics' trends in Twitter can approximate their trends in Google.



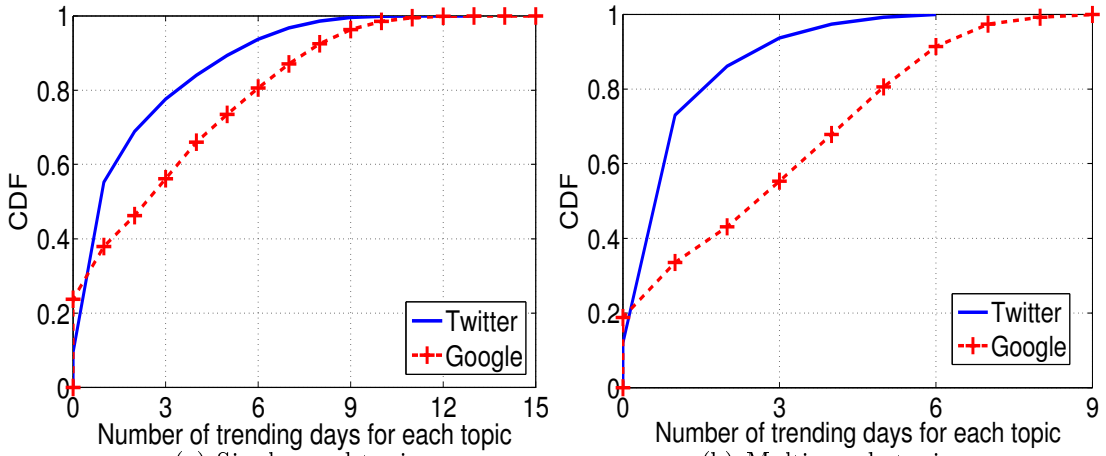
(a) Single-word topics (b) Multi-words topics

Figure 4-1: Kullback-Leibler Divergence between Twitter trends and Google trends

Trends Similarity in Twitter and Google

The work examines the similarity between trends in Twitter and Google using *Kullback-Leibler divergence* (also called relative entropy), which is a measure of the difference between two probabilities X and Y [53]. The definition of K-L divergence is shown as Eq. 2.15. The smaller the K-L divergence is, the closer the two distributions are. In this context, X and Y are related to the Twitter trends and Google trends of topic c, respectively. $X(i)$ (*resp.* $Y(i)$) is the ratio of trending index volume of c at time i in Twitter (*resp.* Google) to the total trending index volume of c observed in Twitter (*resp.* Google).

For each topic that has trends in both Twitter and Google, the work computes the K-L divergence of the topic trends in two spheres. It also computes the K-L divergence of trends for randomly selected topic pairs from two spheres. This random selection is used as null hypothesis. Fig. 4-1 shows the cumulative distribution function (CDF) of K-L divergences. A notable difference between the two K-L divergence distributions for both single-word topics and multi-words topics can be observed. For example, more than 60% of topic pairs have a K-L divergence less than 1 for the same single-word topics, while this percentage is only 43% for random selection.



(a) Single-word topics (b) Multi-words topics

Figure 4-2: Distribution of number of trending days for trends in Twitter and in Google

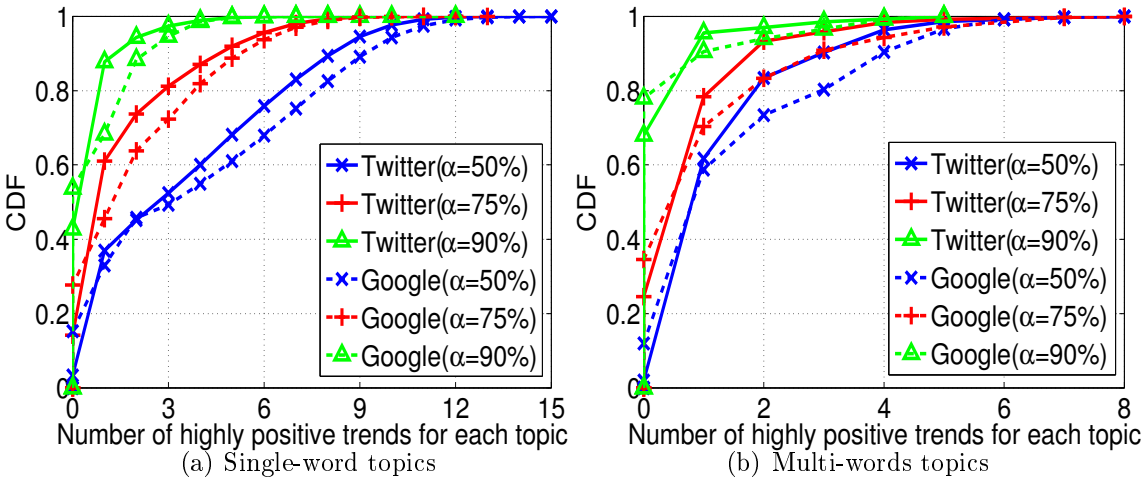
Trending time analysis

The work then examines trending days for each topic (including single-word topics and multi-words topics) $c \in K_d^{\mathcal{T}}$. The number of trending days is defined as the number of days the topics have been tagged as trending (either in Twitter or in Google). Fig. 4-2 shows the distributions of the number of trending days for single-word topics and multi-words topics in Twitter and Google. About 10% of single-word/multi-words topics in Twitter have not been trending (*i.e.* with 0 trending days). This is to be expected because the work only considers Twitter topics that represent a daily set of the most relevant and popular words used in tweets. It can be also observed that about 20% of topics (either single-word or multi-words topics) in Google have not been trending. Recalling that the work uses Twitter topics to crawl Google Trends service, this observation indicates that 20% of these Twitter topics have never been trending in Google.

Interestingly, compared with Google, topics in Twitter have a shorter trending time. For example, about 20% of the single-word topics are trending in Twitter for more than 3 days, while this proportion is 40% in Google and 20% of topics are even trending in Google for more than 6 days. This observation suggests that trendiness of topics in Twitter is much more volatile than in Google.

Highly positive trends analysis

The work then examines for the topics $c \in K_d^{\mathcal{T}}$, the number of highly positive trends they experience. Recalling that a highly positive trend is a positive trend with the increasing trending index volume larger than a threshold α at a particular time, this typically captures a timely and particularly high interest in a specific topic. Here α is varied with 3 typical values: the 50^{th} percentile, 75^{th} percentile and 90^{th} percentile of positive trending index volumes.



(a) Single-word topics (b) Multi-words topics

Figure 4-3: Distribution of number of highly positive trends for trends in Twitter and in Google

Fig. 4-3 plots the distributions of the number of highly positive trends for topics in Twitter and Google. Depending on the value of α, the proportion of Twitter topics that do not exhibit any highly positive trend varies between 10% and 50% for single-word topics and between 10% and 70% for multi-words topics. Google shows a slightly larger number of highly positive trends than Twitter. For example, there are 30% of the single-word topics and 20% multi-words topics hitting more than 2 highly positive trends in Google with $\alpha = 75\%$, while this percentage in Twitter is about 20% for single-word topics and 10% for multi-words topics. The observation indicates that trending topics have a more stable impact in Google compared with in Twitter.

The work further compares the trending days and highly positive days in Twitter

Table 4.2: Comparison of trendiness likelihood in Twitter and in Google for all extracted topics.

Metric		similarity
Trending		65.51%
Highly positive trends	α:50%	69.58%
	α: 75%	51.88%
	α: 90%	33.90%

and Google by checking that whether a similarity exists between them in Table 4.2, where the likelihood is computed as the probability that if a topic which is trending(*resp.* has highly positive trends) in Twitter is also trending(*resp.* has highly positive trends) in Google based on the crawled dataset.

The likelihood that a Twitter trending topic is also trending in Google is 65%, and the likelihood for a Twitter topic that exhibits a highly positive trend with α=50% in Twitter to be similarly showing a highly positive trend in Google is 70%. However, when picking a Twitter topic that has experienced a very highly positive trend (α=90%), there is only 30% of chances for that topic to experience the same highly positive trend in Google. While this lower number potentially stems from the high-selection of such topics in Twitter, it also suggests that Twitter trendiness is potentially more sensitive than Google. Given the different nature of usages of the two services, this is a reasonable explanation as Twitter users would potentially be more reactive to other users interests and topics.

Time offset analysis

The above results call for a deeper investigation of the time effect so that one can understand whether observed trends in one sphere can find their genesis in the other one. For this, the work introduces the *time offset* to represent the difference between the trending times (*resp.* highly positive times) in Twitter and in Google for trending topics (*resp.* topics with highly positive trends). In this study, the *time offset* based on a specific feature of trends (trending or highly positive trends) between the spheres A and B is defined as, the difference between the first day when this feature is observed in sphere A and the first day it is observed in sphere B. A positive value indicates

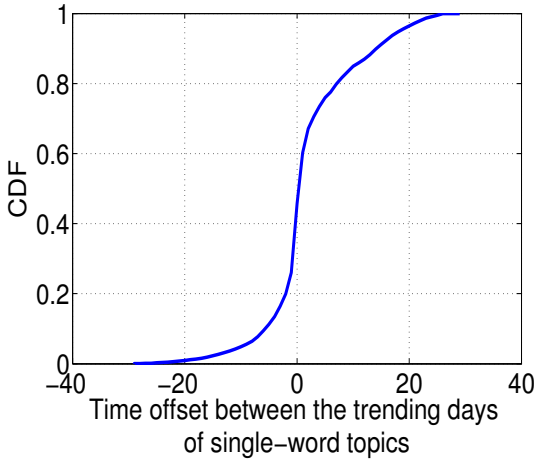


Figure 4-4: Time offset on the trending days between Twitter and Google of single-word topics

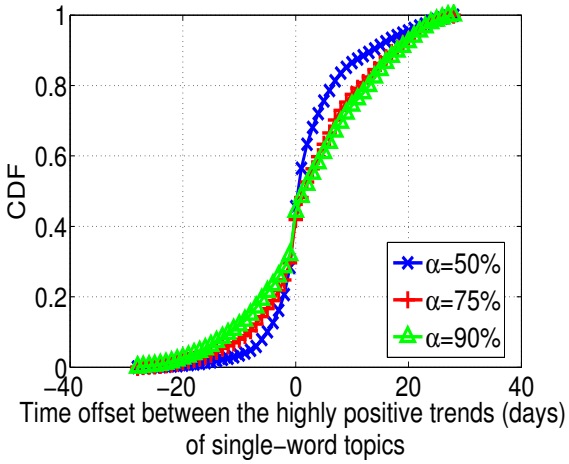


Figure 4-5: Time offset on the highly positive days between Twitter and Google of single-word topics

that the feature happens first in B and otherwise, it happens first in A.

Fig. 4-4 and 4-5 depict the time offsets based on trending days and highly positive days between Twitter and Google for single-word topics respectively. It is clearly observed that most of time offsets assemble around 0 where the proportion of time offsets in [-1,1] interval is much larger than other intervals. In particular, more than half of the time offsets on trending days (*resp.* highly positive days) between Twitter and Google are in [-1,1] interval, indicating that at most a one-day interval separates the trends in these two spheres. There results show that the trendiness in Twitter is likely to be synchronous with the one in Google on daily granularity.

4.4.2 How do Web topics behave in Twitter?

The work now looks at topics extracted from the web sphere, and analyzes their trendiness features in the microblog environment. As mentioned earlier in Section 4.3.2, the Google Trends service unfortunately does not provide enough information on the trending topics in web. As an alternative, the work uses a set of topics extracted from Alexa ranked lists, $K_h^{\mathcal{A}}$. This composes the set of trending web topics. This section focuses on the variation of the "trendiness rank" of topics both in Alexa ($K_h^{\mathcal{A}}$) and in Twitter ($K_h^{\mathcal{T}}$). The analysis in this section can be also based on a finer

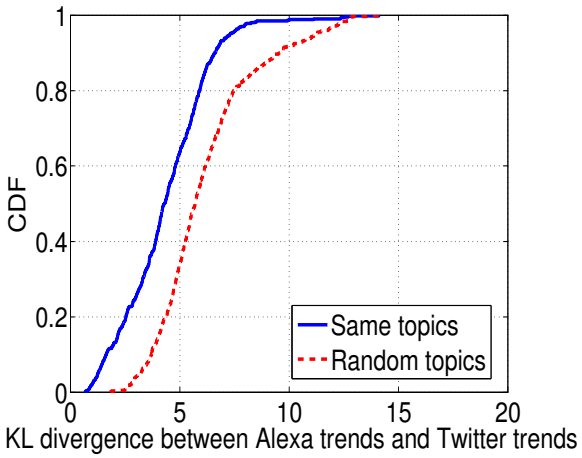


Figure 4-6: Kullback-Leibler Divergence between Alexa trends and Twitter trends

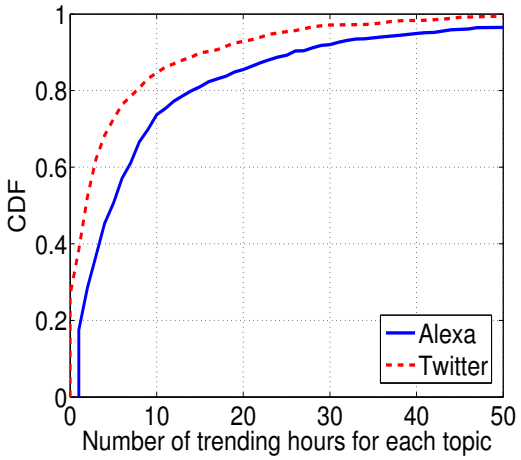


Figure 4-7: Distribution of number of trending hours for Alexa trends and Twitter trends

granularity, *i.e.* hourly as opposed to daily.

Trends Similarity in Alexa and Twitter

Similar to the the analysis of Twitter topics in web, the work also calculates the K-L divergences between Alexa trends and Twitter trends at first. Recall that in Section 3.3.2, it is defined that if topic c is in the top trending list of Alexa at time i, then the trending index volume at i is 100; otherwise, it is 0.

As depicted in Fig. 4-6, if the topic pairs in Alexa and Twitter are randomly selected, the K-L divergences between the two trends are distinctly larger than the ones of same topics, which means the Twitter trends can also be related to the corresponding web trends.

Trending time analysis

Fig. 4-7 shows the trending times (on hourly granularity) of 898 topics $c \in K_h^{\mathcal{A}}$ in Alexa and in Twitter respectively, where the trending hours of a topic c in Alexa are considered as the hours when c appears in the top trending list and the trending hours in Twitter are estimated using the method described in Section 3.3.2. Two notable observations can be got. First, only 28% of Alexa topics have not been trending in

Twitter, which is another evidence that trending topics are similar within the two worlds. Second, topics are likely to be trending for a longer time in Alexa than in Twitter. For example, 16% of topics trending in Twitter for more than 10 hours while the corresponding number in Alexa is about 30%. This observation further confirms the volatility of trendiness in Twitter again.

Time offset analysis

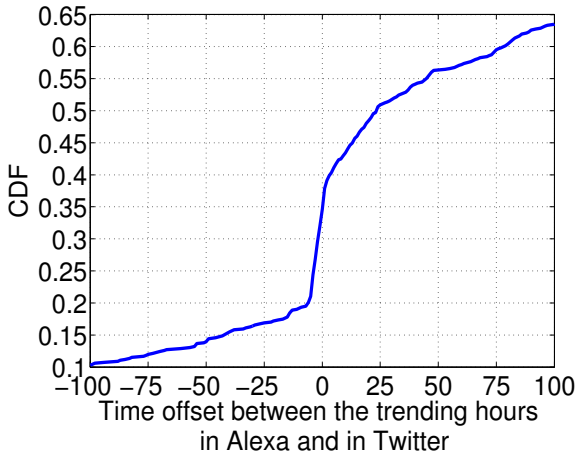


Figure 4-8: Time offset on the trending hours between Alexa and in Twitter

The time offsets (in hour) based on trending times of the same topics between Alexa and Twitter are depicted in Fig. 4-8, where the positive value indicates that the trending feature happens first in Twitter and otherwise, it happens first in Alexa. Opposed to the results in Fig. 4-4, the distribution of time offsets in Fig. 4-8 is skewed towards the positive part, *e.g.* there are more than 65% time offsets are larger than 0 in Fig. 4-8. It can be concluded that although the trendiness in Twitter seems to be synchronous with the one in Google on daily granularity, most of trends of these topics are actually driven by Twitter population in advance, and then spread in web on a finer granularity(such as hourly granularity). This result is also in accordance with the reports in [49].

Rank stability analysis

The work further measures the rank stability coefficient [62] of trends in Twitter and Alexa in order to examine the volatility of trendiness within the two worlds. Given a time frame t, the rank stability coefficient for the top N trending topics in the $i^{th}(i > 1)$ bin is defined as:

$$R_N(i) = \frac{|S_N(i) \cap S_N(i-1)|}{N} \tag{4.2}$$

where $S_N(i)$ is the set of top N trending topics during the i^{th} time frame. The rank stability coefficient has values within [0, 1], where 1 indicates no change and 0 means that all the topics in the list have changed.

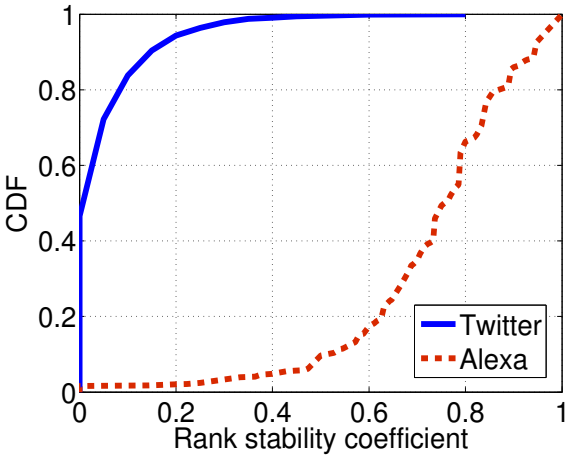


Figure 4-9: Rank stability between Alexa top 20 trending topics and Twitter top 20 trending topics(hourly)

Fig. 4-9 depicts the CDF of the rank stability coefficient of the top 20 (*i.e.* $N = 20$) trending topics based on the topics extracted from Alexa (*i.e.* $K_h^{\mathcal{A}}$) and the topics extracted from Twitter (*i.e.* $K_h^{\mathcal{T}}$) on hourly granularity during the period of August 2009. A notable difference of rank stability coefficient in Twitter and Alexa can be observed. In particular, while there is a limited number of cases in Twitter experiencing a stability coefficients more than 0.5, as many as 90% of the cases in Alexa are more than 0.5. About half of the cases in Twitter have a 0 coefficient, indicating that all the trending topics have changed within one hour. The observations

show the "ephemeral" trendiness in Twitter and much more stable web interests.

4.5 Spatial analysis

The interaction of information spreading in microblogs and web interests is not only reflected in time but also in the spatial dimension. It has been observed in [15] [96] that both the topic's "original" location and the location of the receivers strongly affect the diffusion patterns of the information. This section analyzes the spatial/geographical dimension of the interaction between microblog trends and web interests.

In summary, it is found that the large majority of trending topics appear concurrently in not more than 2 countries in both Twitter and Google, which is a strong evidence of the existence of locality of interest in the trendiness of microblogs and web. Besides, it can be also observed that more than 60% of the locality of interest of individual topics exhibit similar patterns in Twitter and in Google.

4.5.1 Locality of interest

The work introduces the concept of *locality of interest* to characterize the geographic characteristics of trending topics. Five countries are chosen, US, UK, CA, FR and AU, to study whether or not a topic c is trending in a specific location. The fewer number of different regions a topic is trending in, the more significant the locality of interest will be. In order to analyze the locality of interest, the work uses the trending topics provided by Twitter itself from September 1st, 2012 to October 31st, 2012 (dataset H) in these 5 countries, and considers statistics provided by Google Trends for the same topics within the same period.

Fig. 4-10 shows the trending topics overlap in the 5 different countries both in Twitter and Google. It can be observed that the Twitter's trending topics have a more notable geographical concentration effect compared to Google. About 80% of Twitter trending topics appear in only one country while this proportion in Google is only 47.5%. In both Twitter and Google, the majority of topics get trending in not more than 2 countries (95.6% in Twitter and 65.0% in Google). This indicates clearly

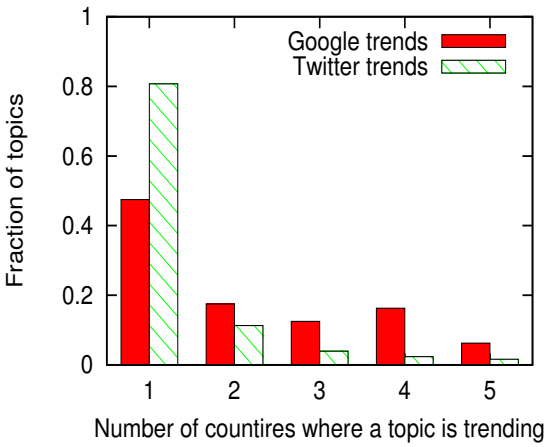


Figure 4-10: The overlap of the trending topics in 5 different countries

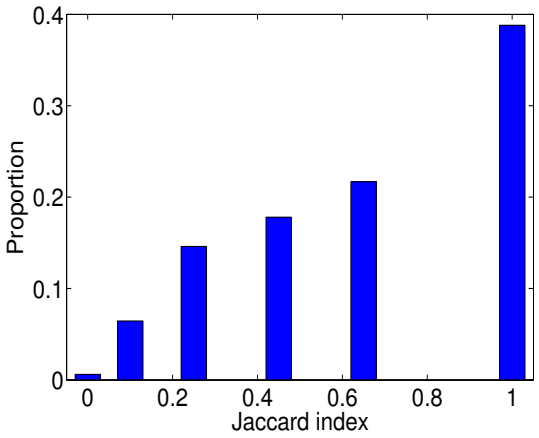


Figure 4-11: Distribution of Jaccard index between interest vectors in Twitter and in Google

that trendiness both in Twitter and in Google are geography-dependent.

4.5.2 Similarity of locality of interest

After confirming the existence of locality of interest in Twitter and Google, the work further checks the similarity of the two spheres in terms of such locality. To this end, the work uses the notion of *interest vector*. The interest vector of topic c is composed of 5 elements in order, $L_{US}(c), L_{UK}(c), L_{CA}(c), L_{FR}(c)$ and $L_{AU}(c)$, each of which is binary and 1 represents the topic c is trending in this country and otherwise the value is 0.

Google Trends provides the top 10 trending countries for each topic, so one can use the appearance in the top list to define the interest vector of Google. That said, $L_r(c)$ in Google is 1 if r is in the Google top country list of c; otherwise, it is 0. As to Twitter, the work focuses on whether a topic is in the top trending topic list for each country. $L_r(c)$ in Twitter is 1 if c is in the Twitter top trending topic list of country r; otherwise, it is 0.

For each topic $c \in H$, there are two interest vectors: $\overrightarrow{L}^{\mathcal{G}}(c)$ for Google and $\overrightarrow{L}^{\mathcal{T}}(c)$ for Twitter. The work computes the Jaccard similarity index of these two vectors for each topic to measure the similarity of Google and Twitter in terms of locality

of interest. The Jaccard index is a statistic used for comparing the similarity and diversity of binary vectors. For two binary vectors $\vec{A}$ and $\vec{B}$, the Jaccard coefficient $J(\vec{A}, \vec{B})$ is defined as:

$$J(\vec{A}, \vec{B}) = \frac{\vec{A} \cdot \vec{B}}{|\vec{A}|^2 + |\vec{B}|^2 - \vec{A} \cdot \vec{B}} \tag{4.3}$$

where $\vec{A} \cdot \vec{B} = \sum_i A_i B_i = \sum_i (A_i \wedge B_i)$ and $|\vec{A}|^2 = \sum_i A_i^2 = \sum_i A_i$. For any pair of vectors $\vec{A}$ and $\vec{B}$, $0 \leq J(\vec{A}, \vec{B}) \leq 1$. The closer this coefficient is to 1, the more similar the two vectors are. Fig. 4-11 presents the Jaccard similarity coefficient for individual topics. It can be observed that more than 60.5% of the topics exhibit a similarity value larger than 0.60(*i.e.* at least 4 elements are the same between the two vectors), which suggests that locality of interest of individual topics exhibit similar pattern in Twitter and Google. In other words, trending topics have similar geographic trends in both Twitter and Google.

4.6 Application

The work has found that individual topics in the Twitter sphere and the web sphere share similar trending patterns from both temporal and spacial aspects. Nevertheless, the trendiness in Twitter can be leading for a few hours and is highly unstable compared to the web. The observations suggest the possibility of inferring trending topics from Twitter for the web sphere, which are traditionally provided by search portals like Google.

In fact, the estimation of trends of queries on search engines (such as Google, Bing *etc.*) is a crucial task in Search Engine Marketing(SEM) analysis. In a typical SEM scenario, advertisers publish their advertisements with the assistant of search engines. In the creation of their advertisements, advertisers choose a keyword or a sequence of keywords (*i.e.* topics in the context of this paper) relevant to their business, called "ad keywords", which will trigger the display of their advertisements in the returned search page of these ad keywords. As such, discovering the ad keywords searched

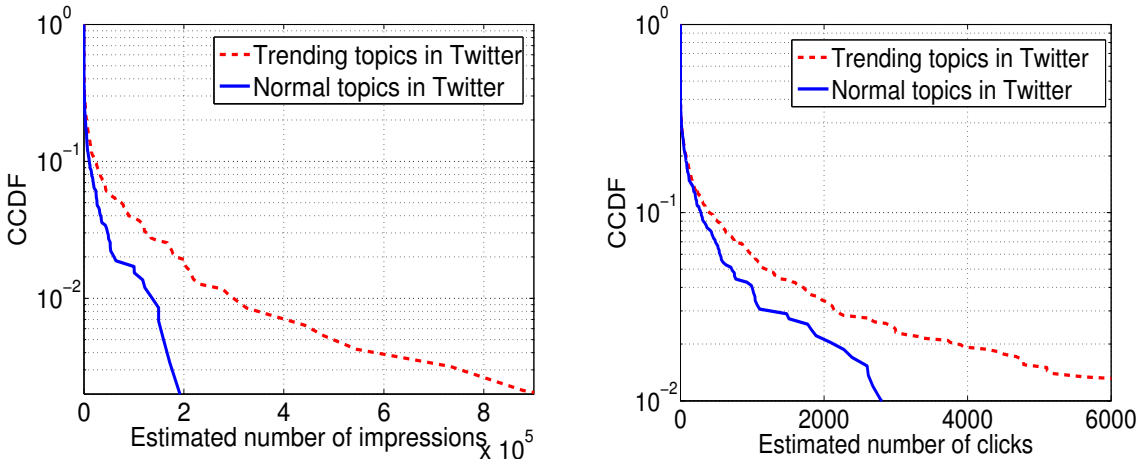
frequently in search engine at a time (*i.e.* trending topics in web) is meaningful to capture high impressions and clicks of online advertisements [48][26][94]. This section shows that trending topics in Twitter could be used to discover superior Google ad keywords.

To this end, the work first samples for every five minutes the top 10 trending topics of Twitter in US during two periods: from October 26th to November 2nd 2013 and from February 2nd to February 8th 2014. This results in a trending topic dataset T consisting of 1,175 unique trending topics. The work also crawls Twitter to get the tweets from US during the same time periods of T using Twitter's streaming API. This results in 105,946 tweets randomly sampled by the Twitter API. Based on these tweets, the work randomly chooses 1,000 words and considers them as a non-trending topics dataset N. These non-trending topics are considered as a reference for this comparison scenario.

For these 2,175 topics obtained from Twitter, the Google AdWords is queried, which provides a "Keyword Planner" tool for helping users evaluate their ad keywords. The input of the tool is the chosen keyword and the output is the estimation of the number of impressions and clicks brought by this keyword based on the previous week statistics [39]. By querying this tool, the work obtains the number of daily impressions and the number of daily clicks of each topic in US for the 10 following days after the topic is sampled from Twitter.

Fig. 4-12 shows the CCDF of the average estimated number of impressions and clicks returned by "Keyword Planner" for trending topics and non-trending topics in Twitter during the considered 10 days. A significant gap is observed between the distribution functions in terms of both impressions and clicks. A high volume of impressions/clicks for the trending topics can be investigated. For example, 2% of the trending topics have more than 200,000 estimated impressions while none of the non-trending topics can reach this volume. The results confirm that the trendiness in Twitter can be used to infer ad keywords with high impressions and clicks in SEM.

Notably, although "Keyword Planner" provides, based on previous week statistics, official estimates for impressions and clicks in Google AdWords platform, obtaining



(a) impressions

(b) clicks

Figure 4-12: The distribution of average estimated number of impressions/clicks in Google AdWords for trending and non-trending topics in Twitter during the 10 days

an up-to-date information about these values is challenging for advertisers simply because of the one week blackout period of "Keyword Planner". However, with the monitoring of Twitter, the results show here that advertisers can figure out the current market "status" of Google AdWords on a fine granularity (hours) basis.

4.7 Summary

This work has compared the trending topics in Twitter and web (*i.e.* Google and Alexa) by considering both the temporal and spacial perspectives, and found that the trending topics in Twitter and search in web tend to follow similar temporal patterns and the trendiness in Twitter can precede by a few hours. However, trendiness is highly unstable in Twitter where top trending lists change more frequently. Besides, there is a geographical concentration effect of interest in both spheres. The trending "localities" are similar in the two spheres as well. Finally, the work shows theses observations can be used for a "smart" predictive choice of ad keywords in SEM.

Chapter 5

The Potential of Twitter in optimization of SEM

5.1 Motivation

Amongst the diverse forms of online marketing and advertising channels (*e.g.* email, mobile advertising), Search Engine Marketing (SEM), where ads are shown along with results of keyword queries, has been the fastest growing channel in the past decade. With up to $19.51 billions, SEM revenues accounted for more than 53% of the total 2012 Internet advertising revenue in North America [46].

In a typical SEM scenario, advertisers are allowed to publish their advertisements (ads in short) with the assistance of search engines in pages returned after search queries. The classic business interaction between advertisers and search engines involves the advertisers paying the search engines when their ads are being shown (Cost-per-Impression payment) or being clicked (Cost-per-Click payment). Search engines implement an online auction for deciding which ads to display in individual returned pages. Advertisers on the other hand bid on sets of keyword relevant to their business, called "ad keywords" or shortly as *"adwords"*, which will trigger the display of their ads in the returned search page for these adwords. Each advertiser maintains an account with a portfolio of adwords, along with a maximum bid value for each adword and an overall daily budget. Typically, the auction implemented by search

engines uses a variant of the Generalized Second Price (GSP) auction mechanism that takes into account the advertisers' constraints, *i.e.* the maximum bid and overall daily budget, along with a Quality Score capturing the relevance of the advertised content to the search query [3]. Note that as advertisers might have different daily budget constraints, the number of times ads displayed per day can be different.

Generally search engines provide to advertisers convenient platforms to manage their ads (such as Google AdWords, Bing Ads *etc.*). However, maintaining a profitable portfolio might still prove challenging for many advertisers, mainly due to lack of time, resources or expertise on ad markets. In this context, "third-party" partners, either advertising agencies (*e.g.* SuperMedia, Web.com), yellow page publishers or freelance consultants, play an important role of intermediaries between advertisers and search engine platforms. The interactions between third-party partners and advertisers create a *secondary market* in SEM (as opposed to the primary market where advertisers directly interact with search engines), where third-parties sell the services of optimizing the advertisers campaigns while advertisers act as service buyers. In the following third-party operators are referred to as "brokers" and advertisers as "customers".

This work presents an economic model of the third-party market in SEM. Based on Google AdWords, a widely-used platform relying on the Cost-per-Click (CPC) payment mechanism, the work first analyzes the economic relations in the secondary Google AdWords market where customers and brokers negotiate their service costs. The analysis shows that in order to optimize his profit while still being able to achieve the customer's demand, a third-party broker should minimize the weighted average CPC of the adwords portfolio.

Specifically, the work develops an optimization framework inspired from the classical Markowitz portfolio management which integrates the customer's demand constraint, and enables the broker to manage the tradeoff between Return On Investment (ROI) and the risk of his adwords portfolio through a single risk aversion parameter. This framework serves as a powerful tool for the broker as it illustrates well the efficient frontier: a curve which gives the optimal Return Over Investment (ROI) for a

given level of risk. The latter is useful for comparing different portfolio construction strategies and to make decision.

An intuitive strategy to maximize the return on investment is to select a (set of) adwords that have low CPC and high potential click numbers. In other words, the major challenge for a broker is then to build and manage such adwords portfolio. In Chapter 4, it has been found that the correlation between Microblog trends and web interests can be used for a "smart" predictive choice of ad keywords in SEM, therefore, the work in this chapter considers Twitter as a possible source of "valuable" adwords. It postulates that by referring to popular and trending topics in Twitter, a broker can foresee a set of adwords that are likely to attract high click numbers, while being not yet detected by other contestants (third-parties and advertisers) and have therefore low CPC. Indeed using adwords extracted from Microblogs is not excluding adwords coming from traditional marketing method, *e.g.*, handpicking adwords by human marketing experience.

In particular, the work verifies that trending and popular topics extracted from Twitter are plausible good candidates to feed the broker's optimal adwords portfolio. More importantly, the work evaluates the application of the optimization framework and shows that a broker could achieve a significant ROI improvement ($\times 4$) over a classical portfolio management, while maintaining the same level of risk.

5.2 Related Work

This section reviews prior work related to SEM optimization and the interaction between Internet activities and financial markets.

SEM Analysis: Several works have targeted advertisers and emphasized keyword optimizations. [48] described the problem of finding relevant adwords. A model for the conversion rate of individual adwords and addressing adwords sparseness was developed in [82]. The authors in [26] proposed an adword suggestion method exploiting semantic knowledge.[33] studied the relationship between adword characteristics, position of the advertisement and the search engine's ranking decision. A

multiword adword recommendation system was developed in [94] and [103] revealed that specific text patterns can lead to high CTR in SEM. Other works have analyzed the user behavior in SEM and proposed mechanisms to maximize the revenue [68][36][11][25][105]. However, most of these target the constitution of the portfolio and no one analyzes the performance of the obtained portfolio and the way to optimize portfolio with Twitter, as this work does.

Financial market analysis in Internet sphere: The study on stock market in [76] found that the high Google search volumes of these terms are always followed by downtrends of DJIA. [29] further analyzed the biases that may affect the backtest of a trading strategy built in [76]. Similar with [76], [71] presented evidences in line with the intriguing suggestion that data on changes in how often financially related Wikipedia pages were viewed may have contained early signs of stock market moves.

Besides from the traditional Internet searching or visiting pattern, some studies have predicted the financial market from the perspective of online social networks. [115] found that emotional tweet percentage is negatively correlated with Dow Jones, NASDAQ and S&P 500. The analysis in [13] indicated that the accuracy of DJIA predictions can be significantly improved by the inclusion of specific public mood dimensions in Twitter but not others. [90] found the sentiment of tweets to be associated with abnormal stock returns and message volume to predict next-day trading volume.[17] confirmed that trending topics offer a comparable visibility to the aforementioned traditional advertisement.

5.3 Analysis of Google AdWords secondary market

This section analyzes the Google AdWords secondary market and uses a simple "Constant Elasticity Demand" based model to capture the broker's profit.

5.3.1 Broker's Profit Analysis

In the context of Google AdWords secondary market, the valuable *product* that a broker provides to his customers consists of the adwords management and the opti-

mization service. In practice, the customer who wishes to maximize the impact of his advertisement campaign, entrusts a third-party broker to build an adwords portfolio. In return he pays the broker a service fee. The broker interacts directly with Google AdWords by choosing relevant adwords, setting the maximum bids while considering the overall budget, and pays the costs of displayed ads to Google. The broker's profit is simply the difference between the service fee paid by the customers and the advertising costs paid to Google.

Different advertisers might have different aims for their advertisement campaigns, *e.g.* reaching a given audience in terms of click number, or achieving a given number of conversions, *i.e.*, clicks that result in other activities like buying the product or signing a petition, *etc.* The broker has therefore to align his profitability objective with the precise needs of his customer.

Typically, two types of contract between the advertiser and the broker are possible: (1) the advertiser has a target objective $D(a)$ for his ad a (a given click number or conversion number) at a time horizon $\mathcal{T}$ and the broker proposes an overall budget; (2) the advertiser has an overall budget for his campaign and the broker commits on the target objective $D(a)$ for this budget at time horizon $\mathcal{T}$. Regardless of the contract type, the efficiency of a third-party broker finally boils down to minimizing the cost paid to Google. However, Google Adwords uses a pay per click model, *i.e.*, the cost paid to Google AdWords depends on the Cost Per Click (CPC), which is determined through the online auction mechanisms and the click number. A reasonable fee strategy for a broker consists then of setting his service fee as function of the click number brought by the customer's ad, *i.e.* the broker sets a Price Per Click or Price Per Conversion (PPC) $P(a)$ for the ad a of his customer. The overall budget can be easily translated to PPC by dividing it to the target objective $D(a)$. Notably, any conversion resulting from a SEM campaign is necessarily bound to a click. That said, one can translate the Price Per Conversion to a Price Per Click by accounting for an average conversion coefficient representing the likelihood that a click results into a conversion, *i.e.*, regardless of the type of contract with the customer and the goal of the customer, the analysis of profitability of the broker entails deriving a PPC $P(a)$

for each click on the customer's ad. As the cost paid to Google is measured in terms of click, for the ease of notation it can be assumed that the target objective $D(a)$ is defined in term of clicks.

The work will therefore assume that the contract between the broker and the customer is set on a Price Per Click (PPC)-basis. The work will also consider that the customer's constraint (demand) in the contract is the number of clicks needed to achieve the objective in terms of clicks or conversions.

In order to satisfy the contract, the broker builds a portfolio of adwords, denoted $K(a)$ for the ad a. This section considers that the adwords portfolio $K(a)$ is *a priori* given and then details the portfolio construction process in Section 5.5. If, for each adword $i \in K(a)$ the $CPC_t(i)$ at time t, as defined by Google AdWords, and a PPC contracted with the advertiser $P(a)$, are given, then the profit of the broker up to time t from ad a can be expressed as the difference between his revenue and costs:

$$Q_t(a) = \sum_{i \in K(a)} \sum_{T(i,a) \leq t} (P(a) - CPC_{T(i,a)}(i)) \tag{5.1}$$

where $T(i,a)$ is the set of time instants when the adword i was searched, the ad a was shown and a click was applied to the ad. While it is intuitive to consider that the high click number tends to result in a high CPC, Yuan *etc.* [114] have found that the bidding of advertisers is always unresponsive to the change of click number, meaning a stable CPC over time. This has also been confirmed by the analysis over the Twitter dataset in Section 5.5.

In this context, $S_t(i,a) = \sum \mathbb{1}_{T(i,a) \leq t}$ is used to represent the number of clicks on ad a resulting from adword i searches up to time t and $S_t(a) = \sum_{i \in K(a)} S_t(i,a)$ to be the total number of clicks on ad a up to time t. With these notations, one can simplify the expression of the broker's profit as:

$$Q_t(a) = \sum_{i \in K(a)} S_t(i,a) \left(P(a) - \overline{CPC_t(i)} \right) \tag{5.2}$$

where $\overline{CPC_t(i)}$ is the average CPC of adword i up to time t. At time horizon $\mathcal{T}$ the

broker will satisfy his contract if the committed demand in the contract is reached and his profit will become:

$$Q_{\mathcal{T}}(a) = \sum_{i \in K(a)} S_{\mathcal{T}}(i,a) \left(P(a) - \overline{CPC_{\mathcal{T}}(i)} \right) \tag{5.3}$$

or equivalently

$$Q_{\mathcal{T}}(a) = D(a) \left(P(a) - \overline{\overline{CPC_{\mathcal{T}}(a)}} \right) \tag{5.4}$$

where $\overline{\overline{CPC_{\mathcal{T}}(a)}} = \frac{\sum_{i \in K(a)} (S_{\mathcal{T}}(i,a) \overline{CPC_{\mathcal{T}}(i)})}{\sum_{i \in K(a)} S_{\mathcal{T}}(i,a)}$ is the weighted average CPC and the double bar notation indicates that the average is calculated both over time and over the adwords portfolio $K(a)$. The Return On Investment (ROI) is therefore calculated as:

$$\overline{\overline{R(a)}} = \frac{\left(P(a) - \overline{\overline{CPC_{\mathcal{T}}(a)}} \right)}{\overline{\overline{CPC_{\mathcal{T}}(a)}}} \tag{5.5}$$

As stated earlier, the above definitions are also applicable for the demands in terms of conversions by accounting for an average conversion coefficient on the customer's click constraint $D(a)$.

5.3.2 The Quality Score

An important element of the broker's profit analysis is the Quality Score, which measures the relevance of the ad content to the search query. The search engine tries to choose the ad that will provide it with the best total estimated revenue. However in order to produce a revenue, an ad has to be clicked by the user searching for the adwords, *i.e.*, the ad and the searched keywords should be relevant to each other. In order to evaluate the relevance between the keyword and the proposed ad, Google defines a "Quality Score" (QS) ranging from 1 to 10 for each pair (adword, ad) that is used in auctions. During the online auction, Google weights the CPC bidden by the advertisers by the QS, and decides therefore the auction winner and the rank of the ad display by combining the CPC and the QS [42].

The algorithm of Google to evaluate the QS is proprietary but it is known that Google uses the past history of clicks happening on a given ad [42]. The latter is a function of both the relevance of the ad content to the adword, and the quality of the ad landing page, *i.e.* the page which a user visits after clicking the ad. This means that the broker can reduce his costs by using adwords portfolio with high QS for his ad. Although the above equations don't have a direct term for the QS, it is considered in Section 5.6 through its impact on the average CPC and on the click number as a better QS leads to lower CPC and more frequent showing of the ad.

5.3.3 Demand modeling

Intuitively, a higher PPC indicates a higher revenue of the broker. However, the price has an immediate impact on the customer's demand. This relationship is generally characterized by a price/demand curve. This work uses a simple customer elasticity model, the Constant Elasticity Demand (CED) model, to describe such relationship. This model assumes that the elasticity of the demand $\eta = \frac{\partial D(a)/D(a)}{\partial P(a)/P(a)}$ is constant, meaning that a relative increase (*resp.* decrease) in the price results in a proportional decrease (*resp.* increase) in the demand with a constant η. The CED model is widely used for describing the user utility on Internet [70]. In particular, it is appropriate for scenarios where the product demands are separable, *i.e.*, changes in demand or price for one product have no effect on others. These assumptions are valid for the Google AdWords secondary market where the demand $D(a)$ only depends on the overall budget and the PPC $P(a)$ on a but not on other ads.

In the CED model, the relationship between the customer's demand $D(a)$ for ad a and the PPC $P(a)$ are described by the following equation:

$$D(a) = \left(\frac{v(a)}{P(a)}\right)^{\alpha} \tag{5.6}$$

where $v(a) > 0$ is a valuation coefficient for a. The parameter $\alpha \geq 1$ is called *price sensitivity* and indicates the price elasticity of demand, *i.e.* $\eta = \frac{\partial D(a)/D(a)}{\partial P(a)/P(a)} = -\alpha$.

The unitary elastic case $\alpha = 1$ happens when the advertiser's budget is constant,

i.e., $P(a)D(a) = v(a) = cte$. However, to make the model more realistic, the work should encompass the case where the customer sets an upper limit for the price $P_{\max}$ and a minimum number of expected clicks or conversions $D_{\min}$. The customer may decide to choose a different broker if the negotiated price is higher than $P_{\max}$ or if the broker cannot satisfy at least $D_{\min}$ objective of his ad. This suggests that a truncated CED model might be a better candidate to describe the customer's demand.

Adding the CED model into Eq. 5.4, the broker's profit becomes:

$$Q_{\mathcal{T}}(a) = \left(\frac{v(a)}{P(a)}\right)^{\alpha} \left(P(a) - \overline{\overline{CPC_{\mathcal{T}}(a)}}\right) \tag{5.7}$$

where $0 < P(a) \leq P_{max}$. The profit-maximizing price $P^{\star}(a)$ for ad a can be obtained by solving $\frac{\partial Q_{\mathcal{T}(a)}}{\partial P(a)} = 0$ and considering the maximum price constraint. It can be expressed as:

$$P^{\star}(a) = \min\left\{\frac{\alpha}{\alpha - 1}\overline{\overline{CPC_{\mathcal{T}}(a)}}, P_{\max}\right\} \tag{5.8}$$

Replacing $P^{\star}(a)$ in the CED model, one can derive the customer's demand needed to achieve the maximal profit as:

$$D^{\star}(a) = \max\left\{\left(\frac{(\alpha - 1)v(a)}{\alpha\overline{\overline{CPC_{\mathcal{T}}(a)}}}\right)^{\alpha}, D_{\min}\right\} \tag{5.9}$$

This demand results in a maximum ROI:

$$\overline{\overline{R^{\star}(a)}} = \min\{\frac{1}{\alpha - 1}, \frac{P_{\max}}{\overline{\overline{CPC_{\mathcal{T}}(a)}}} - 1\} \tag{5.10}$$

5.3.4 The rationale for adwords portfolio

The above analysis shows that when the broker has an a priori knowledge of the number of clicks, the conversion rate and the average CPC at a time horizon $\mathcal{T}$ for each adword i in the ad a (*i.e.* the weighted average CPC of a at $\mathcal{T}$ $\overline{\overline{CPC_{\mathcal{T}}(a)}}$), the broker can easily optimize his profit by setting the selling PPC slightly higher with a coefficient $\frac{\alpha}{\alpha - 1}$ than the weighted average CPC.

However in practice, the Google AdWords market is very dynamic. The instantaneous value of CPC for an adword, the number of clicks and the conversion rate vary at the whim of auctions and search engine users' willingness (or interest) to click on the displayed ads. One approach is then to cover for the risk of variation of all these parameters by considering aggregation of adwords in a portfolio in place of using a single adword.

5.4 Dynamics of adwords Portfolio

The instantaneous value of the CPC for an adword, the number of clicks and the conversion rate are stochastic processes which vary with time, meaning that ROI should be considered as a random variable with mean $\mathbb{E}\{\overline{\overline{R(a)}}\}$ and variance $\sigma^2(\overline{\overline{R(a)}})$. For an ad a attached with a portfolio of adwords $K(a)$, the average and variance of ROI can be derived as:

$$\begin{cases} \mathbb{E}\{\overline{\overline{R(a)}}\} & = \sum_{i \in K(a)} w_i R(i) \\ \sigma^2\left(\overline{\overline{R(a)}}\right) & = \sum_{i,j \in K(a)} w_i w_j \sigma\left(R(i)\right) \sigma\left(R(j)\right) \rho(i,j) \end{cases} \tag{5.11}$$

where $w_i = \frac{S_{\mathcal{T}}(i,a)}{D(a)}$ is the weight of adword i in the portfolio, *i.e.*, the proportion of clicks or conversions resulting from the adword $i \in K(a)$ among all clicks or conversions leading to the ad a satisfying the demand, $R(i)$ is the ROI of adword i and $\rho(i,j)$ is the correlation coefficient between the ROI of i and j.

Again, the overarching objective of a broker is to maximize his ROI by satisfying the customer's demand $D(a)$ on an ad a, while minimizing the stochastic risk resulting from market fluctuations. In such a context, the risk for the broker is that the final $\overline{\overline{CPC_{\mathcal{T}}(a)}}$ becomes larger than $P(a)$ resulting in loss. In order to protect the broker from such a risk, the work will adopt an approach inspired from the Markowitz formulation of portfolio optimization in financial market [67]. The Markowitz portfolio optimization defines the proportion of capital that an investor should dedicate to different assets with different ROI and risks, in order to maximize his profit given a

level of risk aversion. In the approach of this work, the stochastic risk of the broker is captured by $\sigma^2\left(\overline{\overline{R(a)}}\right)$ and his aversion to risk is characterized by a value $\gamma > 0$ named risk aversion coefficient. The larger γ is, the more risk the broker is ready to take in order to increase his ROI. In this case, the stochastic version of the broker optimization problem can be written as:

$$\min_{\mathbf{w}\in\mathbf{\Delta}} \left(\sigma^2\left(\overline{\overline{R(a)}}\right) - \gamma\mathbb{E}\{\overline{\overline{R(a)}}\}\right) \tag{5.12}$$

where $\mathbf{w} = (w_i), i \in K(a)$ is the vector of weights and Δ is the simplex surface $\{\mathbf{w} \in [0,1]^{|K(a)|} | \sum_{i\in K(a)} w_i = 1\}$.

The major difference between the classical Markowitz formulation and the formulation used in this work comes from the constraint to achieve the customer demand $D(a)$ rather than only trying to maximize the portfolio ROI as in classical Markowitz. In other terms, the Markovitz formulation with demand constraint case is used here. Moreover, in classical Markowitz formulation the share of the capital assigned to each asset w_i is a deterministic value that is set at the time of the constitution of the portfolio, while in the formulation here the w_i is a random variable depending on the willingness of search engine users to engage with the ad, and on the relevance of the ad content to the adword i. This means that over a fixed time horizon, the value of w_i may be smaller or larger than the optimal value. This difference becomes more important when the customer's demand is described through a conversion number, because this adds one more element of randomness: the decision of the viewer to convert a click into a concrete action like buying the product.

The metrics randomness can be dealt with in three ways. The first approach consists of relaxing the time horizon, *i.e.*, the value w_i can be considered as deterministic and the adword i is kept using till it attains the target share w_i. The second approach takes some cautions with the official click number estimates provided by Google AdWords, *e.g.*, limits the value of w_i to a percentage $0 < \beta \leq 1$ of the click number estimated during the optimization, *i.e.*, $w_i < \frac{\beta S^g_{t_0}(i,a)}{D(a)}$ where $S^g_{t_0}(i,a)$ is the click number estimate given by Google AdWords at the time of decision t_0. This

adds a new constraint to the optimization problem defined in Eq. 5.12. However, this approach is only applicable when the customer's demand is expressed in terms of audience size (clicks). Such an estimate is not available for the conversion demand at the beginning of an ad campaign as the conversion rate of this ad is unknown. The third approach that is also applicable when the target demand is a conversion number, is a dynamic version of the second approach, where the weight w_i for each adaptation period (*e.g.* each day) is limited by the number of clicks or conversion rate observed so far in the previous observation periods. The conversion rate can be observed in real time through a script in checkout page that accounts for each finalized transaction resulting in an adword click. It is noteworthy that the weights are unlimited during the first period where previous observations are missing and this entails a re-optimization process at the beginning of each period.

A useful concept of portfolio management is "efficient portfolio". A portfolio is called "efficient" if it has the best possible expected ROI for its level of risk, $\sigma^2\left(\overline{\overline{R(a)}}\right)$. The efficient portfolio is illustrated through the risk/ROI plane, a frontier separating achievable risk/ROI tradeoffs (on the right of the curve) from the unachievable one (on the left of the curve), as shown in Fig. 5-1. The points on the efficient frontier can be calculated by solving the optimization in Eq. 5.12 for different values of risk aversion γ and plotting the resulting optimal ROI and risk. In the Modern Portfolio Theory, this efficient frontier is always used as a metric to compare different portfolio constitution approaches. A top-left oriented frontier means that higher ROI with lower risk is achievable. This frontier is utilized later for evaluation.

5.5 Building the Portfolio

So far, the work presented the theoretical foundations for optimizing an adwords portfolio by a third-party broker, where the portfolio is assumed to be a priori given. However, building an adwords portfolio in practice is far from being trivial because of the dynamics of adwords market. This section first proves an overview of possible ways to constitute a portfolio and then proposes one approach to augment portfolios with

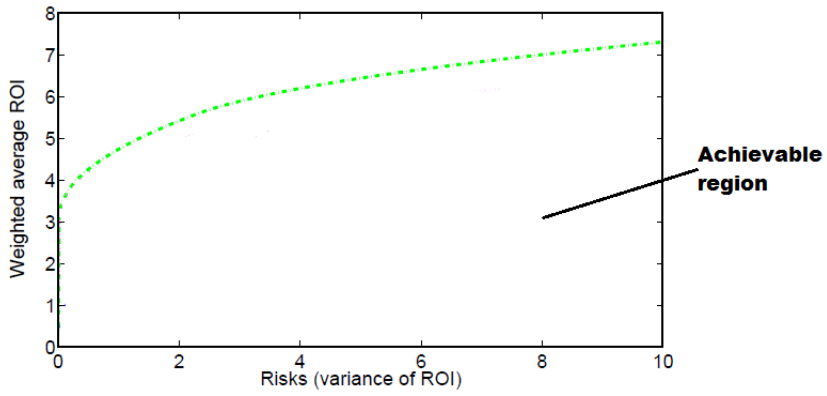


Figure 5-1: An illustration of efficient frontier

additional adwords. Unless stated otherwise, the customer's demand is considered to be expressed in terms of number of clicks.

Google provides a "Keyword Planner" tool for helping users choose and match adwords [39] to their ad campaigns. Through this interface users can select and test several combinations of adwords portfolio. Using "Keyword Planner", the aim of the broker would be to uncover the adwords with low CPC and high potential of clicks or conversions. However, such adwords are likely to attract competition quickly and their CPC are likely to increase fast in the future. Generally, two approaches might be considered in this context. The first approach consists of searching for "long-tail" queries, *i.e.*, infrequent queries that are likely to draw targeted visitors on ads, *e.g.* the bulk of Amazon's revenues comes from a long tail of items but not from a few block-buster items [7]. Several business sites are targeting such keywords in search engines recently [88][10]. The second approach consists of exploring the adwords space for promising topics which have not yet attracted the interest of competitors (third-party brokers and advertisers), but have already generated a surge in search traffic and as such are likely to be efficient from a user interest perspective. This aims not at replacing the first approach but rather at augmenting it.

The work considers the second approach as one possible strategy to build and augment the adwords portfolio. Previous researches have revealed how stock market changes can be predicted based on observations of Twitter trends [115][13][90][17]. Therefore this work considers topics originating in Twitter as potential candidate

adwords for an efficient portfolio. In the following, the work describes this method to extract popular and trending topics from Twitter to feed the adwords portfolio construction and analyzes the Google AdWords properties of these extracted topics.

5.5.1 Twitter Data Collection

Twitter Dataset

Any word or sequence of words mentioned in tweets can be considered as a potential *topic*, i. The popularity of a topic i can be defined as the number of tweets mentioning it. Topics mentioned relatively more frequently over a time period are called *popular topics*. *Trending topics* on the other hand are defined by Twitter as topics with a popularity that is increasing relatively faster than other topics [107]. Trending topics emerge either endogenously, driven by evolution of users interests, or exogenously, *i.e.* caused by external events that prompt people's attention. A "non popular" topic might be "trending" when its popularity increase rate is large. Moreover a topic remaining popular for a long time is not likely to stay "trending", as the number of tweets mentioning this topic stabilizes. In order to study the relation between Google AdWords properties and Twitter popularity, the topics extracted from Twitter are stratified into three classes: trending, popular and normal (*i.e.* random).

Trending topics: The work extracted the topics provided publicly by Twitter as trending over the period spanning from October 26th to November 1st 2013 and from February 2nd to February 8th 2014. Specifically, every five minutes during the crawling period, the work collected the top 10 trending topics in US suggested by Twitter. It is noteworthy that a topic can be trending for more than one day. In such case, the work only considered the trending topics which had never been trending before the sampling day. Finally, this resulted into 1,175 unique trending topics that composed the trending topics dataset T. The reason why the two time periods are used is to catch the trending topics related to candy(*e.g.* "Halloween") and sports(*e.g.*"super bowl" and "world series") respectively, which will be used for

analysis and evaluation later.

Popular topics: Popular topics were more challenging to extract as one had to crawl and sample original tweets. The work used the Twitter streaming API to crawl a set of tweets over the same periods as above. This resulted into 105,946 tweets randomly sampled by Twitter API. Then all sampled tweets were binned into subsets with daily granularity and for each subset, the set of words W_i was extracted in order to compute the word frequency in this daily subset for each word $w \in W_i$. Note that stop words (*e.g.* "a", "after", "that", *etc.*) which naturally appeared with higher frequencies were ignored and a word was counted only once per tweet even if it was repeated in the tweet. The hashtags consisting of more than three words were also filtered out as they were too long to make adwords. After the data preprocessing, the work chose the top-200 most frequent words for each daily subset, leading to 2,800 popular topics out of 35,705 topics extracted over the two weeks. Again, the work also removed the duplicate topics and only considered the topics which had never been categorized as popular before the sampling day. This resulted into 1,214 unique popular topics and constituted the dataset P. Although only 3.4% of potential words (topics) were chosen, these represented more than 30% of the total popularity (in terms of volume of tweets mentioning them) over all extracted words.

Normal topics: From the same set of sanitized tweets crawled for popular topics, the work also randomly chose 1,000 other words and considered them as the normal (random) topics dataset N. This set will be used as a comparison with the two others.

Google AdWords Extraction

The work used the "Keyword Planner" tool of Google AdWords to collect, for all 3,389 topics in the three Twitter datasets, the daily CPC and number of daily clicks estimates for the 10 days following the first day each topic was considered as trending/popular or randomly sampled from Twitter.

As reported in [76], the metrics provided by Google slightly change over time due to Google's extraction procedure. To take this into account, the work sampled for each day 8 time points (once every three hours) and then used the average value over

these samples as a single daily metric for each adword. All CPC values are in US dollars. As the display of an ad in pages returned by the search engine depends on the results of an auction run for each displayed page which is relevant to the max CPC bid set by the broker, the larger this value the more likely to win the auction. In order to reduce the randomness and uncertainty related to the auction, the work set daily budget and max CPC bid to the maximum values allowed by Google so that the estimates returned by Google were the estimated maximum CPC which could win all auctions, as well as the estimated maximum click number. This means that the ROI obtained are lower bounds, *i.e.*, the broker can hope to achieve higher ROI than what is reported in this paper.

The words with at least one non-zero CPC value in the 10 days account for 50% of the topics in the normal topics dataset, for 68% in the popular topics dataset and for 63% of the trending topics. Equivalently, 50% of normal topics (*resp.* 32% of popular topics and 37% of trending topics) have never been used as adwords. These null-value words are not considered in this study since they are inactive in Google AdWords and as such it is unable to evaluate the process of using them.

5.5.2 Analysis of the Twitter topics

Table 5.1 shows the relevant statistics derived for topics that have at least a non-zero CPC in the 10 days of Google AdWords monitoring, *i.e.*, topics that are active in Google AdWords.

The statistics show that the daily average CPC and the corresponding variance of the CPC are very stable across the three datasets. A non parametric Kolmogorov-Smirnov distribution test used could not reject the hypothesis that the three datasets come from the same distribution (using a 5% significance level). However both the distributions of average CPC and variance of CPC are highly skewed as the medians are far from the averages. The unbalance is mainly due to the tail, that is, some very large values pull the mean away from the median.

The daily number of clicks shows different distribution statistics across the three datasets. Clearly the estimate of daily number of clicks for popular topics is larger

Table 5.1: Google AdWords properties of topics in the three Twitter datasets

Dataset	Average	Variance	Median	1-prct	5-prct	95-prct	99-prct
$\overline{CPC(N)}$	3.52	22.71	2.15	0.001	0.02	11.88	24.63
$\overline{CPC(P)}$	3.54	19.47	2.30	0.004	0.05	10.82	20.59
$\overline{CPC(T)}$	3.42	19.40	2.23	0.004	0.08	12.93	25.42
$\sigma^2(CPC(N))$	1.56	4.07	0.86	0.004	0.02	7.58	16.26
$\sigma^2(CPC(P))$	1.73	7.96	0.80	0.008	0.04	7.32	14.95
$\sigma^2(CPC(T))$	1.86	8.95	0.87	0.008	0.08	6.33	9.97
$\overline{Clicks(N)}$	329.7	7.8×10^5	35.9	0.01	0.10	1866	5166
$\overline{Clicks(P)}$	928.9	1.5×10^7	51.3	0.03	0.25	4490	18332
$\overline{Clicks(T)}$	476.0	3.5×10^6	34.49	0.06	0.26	1934	7841
$\sigma^2(Clicks(N))$	109.0	4.8×10^4	23.3	0.03	0.0.17	554	1285
$\sigma^2(Clicks(P))$	224.5	6.4×10^5	25.1	0.07	0.29	904	4390
$\sigma^2(Clicks(T))$	162.1	3.2×10^5	20.91	0.07	0.31	547	3445

than the other two. Interestingly, the comparison of normal and trending topics shows that while the medians are the same, the average estimate of the number of clicks of trending topics is significantly larger than the normal ones, indicating that there are more topics in the tail of trending topics with very large number of clicks. This observation combined with the fact that there is no significant difference in the CPC value amongst the three strata is very promising. In fact, this suggests that with similar CPC values (prices), the broker can expect a larger average number of clicks for popular and trending topics. According to the analysis in Section 5.3, the $\overline{\overline{CPC_{\mathcal{T}}(a)}}$ controls the ROI of ad a, therefore a higher number of clicks for an adword with a stable CPC means a lower $\overline{\overline{CPC_{\mathcal{T}}(a)}}$ for the broker and a higher profit.

Lastly it can be observed that the variance of click number estimates shows a significant difference between the normal stratum and the other datasets. The variability in terms of click numbers for popular and trending topics is higher than normal topics. In order to evaluate whether this should be interpreted as a higher risk or a higher opportunity for popular and trending topics, the work also analyzes the estimate of clicks increase. Specifically, for each adword, the work fits the 10 daily click estimate values into a linear regression function and extracts the growth rate of the estimate click number as the slope of the fitted function. Table 5.2 provides the relevant statistics which show that the higher variability is in fact an opportunity, as the number of clicks for popular and trending topics is growing faster than the

Table 5.2: Clicks growth for the three datasets

Dataset	Average	Median	95-prct
Growth rate in N	5.14	1.19	64.74
Growth rate in P	10.04	1.16	182.45
Growth rate in T	11.37	1.29	252.42

normal topics. It is noteworthy that the medians for normal and popular topics are very close and again the tails of popular and trending topics are the key difference.

In summary, the analysis of the three datasets over Twitter shows that while there is not a significant difference in the CPC of adwords originating from Twitter topics, there is a major benefit in terms of number of clicks to add these adwords in the portfolio.

5.5.3 Portfolio constitution methodology

The great potential of the popular and trending topics from Twitter in improving the ads clicks motivates the following portfolio constitution methodology. In detail, a broker follows two steps to build an efficient portfolio. At first, he generates an initial reference portfolio using either adwords suggested by Google (*e.g.* via "Keyword Planner"), or any of the numerous methods developed in the past couple of years for adwords portfolio selections [48][26][94], or even random adwords portfolio selections. In the second step, the broker looks at trending and popular topics coming from Twitter and augments his reference portfolio with relevant trending and popular topics. In other words, the work is aiming not at replacing the existing methods, but rather at augmenting them with topics from Twitter. It is noteworthy that the objective in this work is not to evaluate the initial reference portfolio selection itself but rather to show a portfolio augmentation technique and to suggest the interest of adding adwords coming from Twitter popular and trending topics. As such the work does not compare nor describe these advanced methods of adwords selection but rather simply use Google AdWords suggestions and random adwords portfolio selection.

The methodology is best explained by two case studies: an online candy seller ad and a sports apparatus e-shop ad, both of which are assumed to contact a broker

to start up Google AdWords campaigns. The work uses the "Keyword Planner" of Google for initial reference portfolio selection as in addition to price estimation of adwords, the "Keyword Planner" can also provide a set of suggested adwords for a specific product. The work makes use of these two functions of "Keyword Planner" to find relevant adwords for ads and build an initial reference portfolio. The fact that the initial reference portfolio is derived using "Keyword Planner" ensures that adwords in the reference portfolio have estimates (coming from Keyword Planner) for the average daily CPC, the variance of the CPC and the click numbers. Using these values, the broker first checks the range of click numbers for which the reference portfolio is feasible, *i.e.*, the set of click number which he can commit to satisfy his customer's demand. He thereafter derives the maximum average CPC with minimum risk $\overline{\overline{CPC^{+}(a)}}$. This latter value is obtained by deriving the optimum portfolio with risk aversion $\gamma = 0$. $\overline{\overline{CPC^{+}(a)}}$ is then used to set the price $P(a)$.

For each of these two scenarios it is assumed that the first day in the dataset is the decision day for the broker, and the broker generates an initial portfolio of adwords at the decision day. This initial portfolio is used for two purposes: to set the price $P(a)$ negotiated with the customer by broker and to be a reference portfolio compared with other strategies. Thereafter, the broker looks at trending and popular topics and chooses some of them to augment his portfolio.

Specially, to build the initial reference portfolio, the broker first queries "Keyword Planner" to get the top-5 suggested adwords. This portfolio contains "candy", "online", "chocolate", "bar", "shop" for the candy ad and "sport", "ball", "football", "baseball", "ride" for the sport ad. The work augments such reference portfolio with 5 relevant topics extracted from the trending and popular topics in the Twitter dataset, *i.e.*, "halloween", "halloween images", "halloween quotes", "happy halloween", "trick or treat" for candy ad and "nba", "clippers", "real madrid", "world series" and "super bowl" for the sport ad. The portfolio constitutions of the two ads are shown in Table 5.3. As the data gatherings happened in November and February, the trending topics mentioning "Halloween" for the candy ad and mentioning "world series" and "super bowl" for the sports ad can be got.

Table 5.3: Adwords used in the two specific scenarios

Candy seller		Sports apparatus e-shop	
Google	Twitter	Google	Twitter
candy	*halloween*	*sports*	*nba*
online	*halloween images*	*ball*	*clipper*
chocolate	*halloween quotes*	*football*	*real madrid*
bar	*happy halloween*	*baseball*	*world series*
shop	*trick or treat*	*ride*	*super bowl*

The work evaluates the reference versus augmented portfolios for the above two scenarios in Section 5.6. Naturally, two particular scenarios are not enough to validate the approach. The work thus also utilizes the random adwords selections as the initial reference portfolio constitution method, a technique frequently used in stock market studies. Again, it is noteworthy that the adwords augmentation using popular and trending topics in Twitter is independent of the initial portfolio constitution methods.

5.6 Applications and evaluation

This section evaluates the adwords portfolio constitution method using the optimization model described in Section 5.4. The aim is to evaluate whether the portfolio management methodology developed is able to achieve high ROI with low risk.

5.6.1 Evaluation methodology

First, the challenges faced to make a meaningful evaluation of this research is presented. Two broker companies (Adobe, 4-traders) are contracted and these were not willing to provide details about their methodologies of selecting adwords as obviously these were trade secret. However, none of these contacts was aware of the analytic portfolio management technique like the one proposed in this paper. In this context, the work has no baseline method used in practice to compare with. It therefore resorts to simulating the application of the portfolio to the Google AdWords market by assuming that the estimates provided by Google AdWords for click number and CPC are reliable, *i.e.* the work will assume that during the days after the decision time

t_0 the number of views and the CPC for each adword will be as returned by Google AdWords and the evaluation of a portfolio will be derived using these data.

A critical aspect of this evaluation is the reliability on statistics estimated (provided) by Google. As the researchers are not acting directly on the Google AdWords market there is no way of verifying Google's data reliability. However, as described earlier the work takes a conservative approach by setting the daily budget and max CPC bids to the maximum values allowed by Google in order to obtain higher bounds on CPC and click numbers. This ensures that the ROI derived in this paper represents a lower bound. Moreover, as the customer's clicks demand is likely to be less than the overall capacity of clicks, the broker can stop bidding on an adword when he reaches the adword-clicks objectives determined by the portfolio management optimization.

As explained in Section 5.3, the final CPC depends on the Quality Score (QS) that is variable (and unknown for the broker at the beginning of a campaign). This implies that "uncommon" adwords which potentially are of low relevance to the ad in consideration, might be of low CPC value. Such value in turn is expected to change in time. The broker has then to re-optimize the portfolio periodically to account for changes that are induced by QS variations.

In order to account for QS variations of trending and popular topics that are likely to be less relevant to the ad at the beginning of the campaign (time t_0) than the adwords suggested by Google, the work takes a conservative approach which assumes that the number of achievable clicks on trending and popular topics is no more than half the value reported by Google AdWords. This is equivalent to setting $\beta = 0.5$ in the click number constraints defined in Section 5.4. This assumption ensures that the obtained ROIs are likely to be lower than the actual values that would be observed in practice. Moreover, as the work is not operating as a real broker with actual ads, conversion can not be observed. As such the work cannot evaluate the case where the customer's demand is expressed in terms of conversion numbers, and the following evaluation exclusively considers the customer's demand expressed in terms of click number.

In these experiments, each ad at most has 10 adwords (as suggested by Google

[1]), *i.e.* $|K(a)| \leq 10$ and the customer demand curve is compatible with a CED model where $\alpha = 2.5$. The PPC charged to the customer can be derived from Eq. 5.8 as $P(a) = 1.67\overline{\overline{CPC^{+}(a)}}$ to ensure maximal profit for broker.

In order to define and set a practical scenario the work will assume that the demand of the customer for his campaign is 500 clicks per day (this number of clicks per day is in accordance with values reported in [91]). In the forthcoming the work will apply the portfolio management approach developed in Section 5.4 to derive the efficient frontiers for the reference portfolio along with the augmented portfolio. The portfolio compositions of the two portfolios for a risk of 1 are derived and compared. The performances of the augmented portfolio are also calculated and compared with the reference portfolio. The work will also derive the efficient regions and the ROI respectively with and without the click number constraint, *i.e.*, without guarantee to attain the target clicks number resulting in the classical Markowitz portfolio case, for each portfolio (reference and augmented).

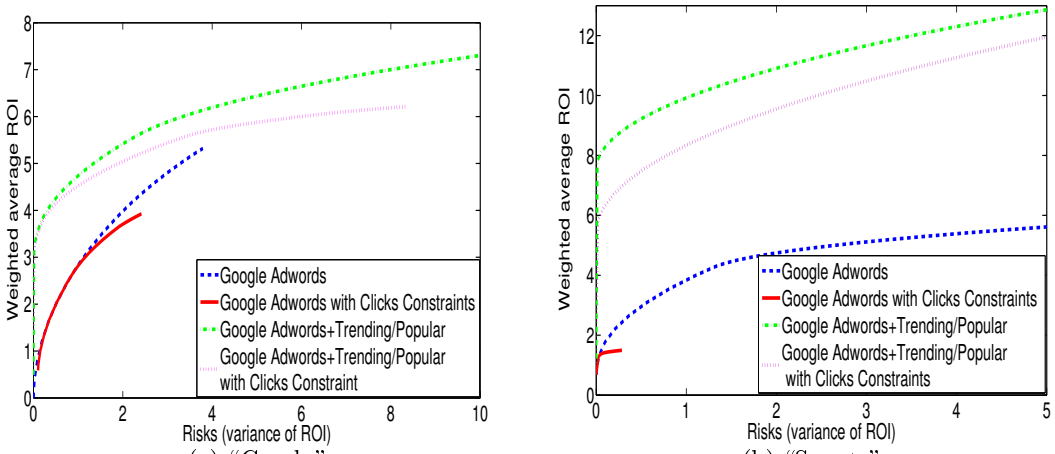
5.6.2 Portfolio performance analysis

The work applies the above methodology to the two toy examples (Candy and Sport apparatus stores) and also to the random initial portfolios.

Two ad cases studies

This section first derives $P(a)$ price for Candy ad (*resp.* Sports ad). The reference portfolio achieving the 500 clicks per day demand with the lowest risk (derived as Section 5.6.1) attains $\overline{\overline{CPC^{+}(a)}} = 3.70$ USD per click for Candy ad (*resp.* 4.12 USD per click for Sports ad). This results in a selling price of $P(a) = 6.18$ USD per click for Candy ad (*resp.* 6.88 USD for Sports ad).

Fig. 5-2 shows four efficient frontier curves for the two specific scenarios respectively, depicting the largest expected ROI for a given level of risk. The region of achievable (ROI, risks) pairs for which there exists a portfolio that can achieve this ROI with the given risk, is the set of points on the right and below the efficient



(a) "Candy" (b) "Sports"

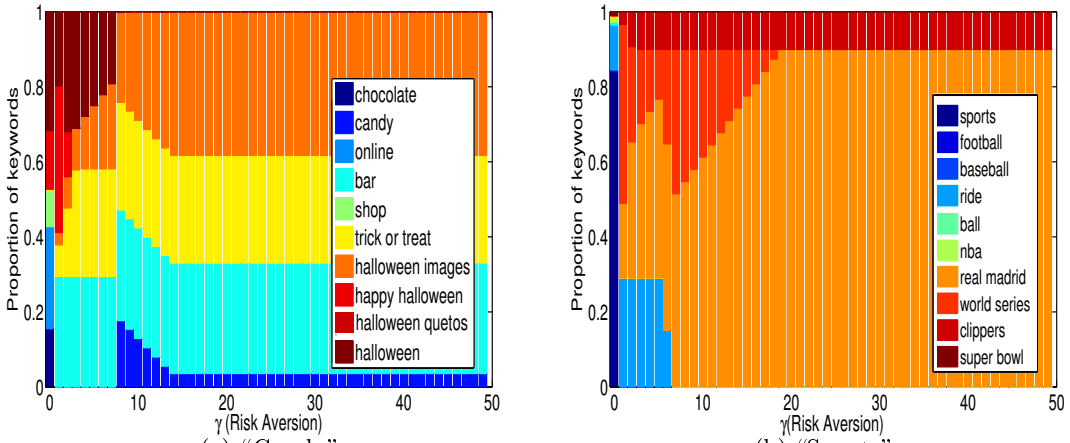
Figure 5-2: The efficient frontier of the two specific scenarios

frontier curve.

As expected the click number constraint reduces the achievable region area. For instance, in Candy scenario, the ROI of portfolio using Google AdWords with click number constraint can only reach 3.92 with a risk of 2.42, while the largest ROI of the portfolio using Google AdWords without click number constraints is 5.03 with a risk of 3.93. The constrained Google AdWords portfolio of Sports apparatus ad spans a very small range of ROI as the largest ROI is 1.49 with a risk of 0.28. This can be explained by the additional restrictions the click number constraint brings to the optimization model as there are only lesser number of adwords that can provide enough clicks to achieve the necessary demand.

Nonetheless the trending and popular topics largely extend the reachable region by augmenting the portfolio with adwords that seem to be more likely to meet the click number constraint. For example, in the Candy ad, to achieve the same ROI of 3.92, the augmented portfolio experiences a risk of only 0.34 while this risk of using Google AdWords is as high as 2.42. The augmented portfolio can even achieve a ROI as large as 5.95 but with an associated risk of 8.20.

Fig. 5-3 further shows the keywords composition of the optimal portfolios for different values of risk aversion γ where the risk level is equal to 1. For lower values of γ, the portfolio contains a larger share of adwords suggested by Google to benefit



(a) "Candy" (b) "Sports"

Figure 5-3: The portfolio composition of the two specific scenarios

from the average risk reduction effect of portfolios. With larger values of γ, the portfolio evolves towards a larger share of trending and popular topics (referring to Table 5.3).

The observations of the resulting portfolios from Fig. 5-3 also show that despite the possibility of utilizing the 10 adwords in the augmented portfolio, all efficient portfolios just use 3 or 4 adwords. For example, it can be found that for a large range of risk aversion only 3 adwords ("real madrid", "clippers" and "world series") remain active in the "Sports apparatus" scenario. This shows that the higher performance of the augmented portfolio is not only due to the mechanical effect of the adword augmentation, but rather to the quality of the additional adwords.

Note that in practice it is necessary to account for the effect of the quality score, QS (or eventually the conversions number) by re-estimating the average and the variance of CPC and click number on a daily basis. There is then a need to "update" the parameters of the optimization model. It is found that the optimization process for a small portfolio (*e.g.* the two examined scenarios in this section) is executed in less than 2 seconds, so the time cost is not a major issue.

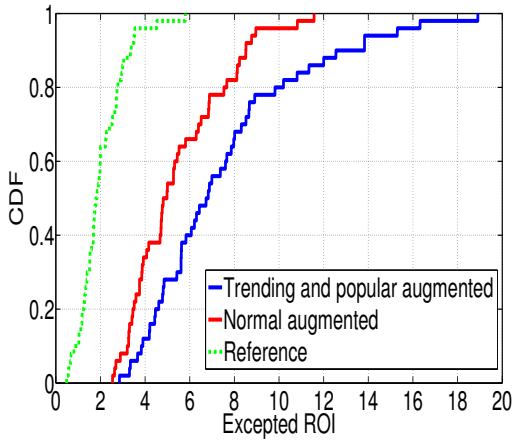


Figure 5-4: CDF of $\overline{\overline{R(A)}}$, $\overline{\overline{R(B)}}$ and $\overline{\overline{R(C)}}$

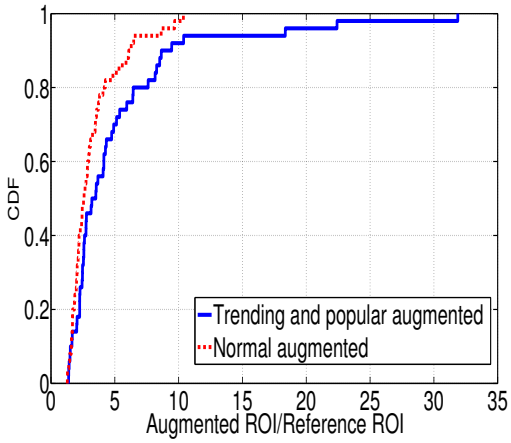


Figure 5-5: CDF of $\frac{\overline{\overline{R(B)}}}{\overline{\overline{R(A)}}}$ and $\frac{\overline{\overline{R(C)}}}{\overline{\overline{R(A)}}}$

Random Portfolios analysis

Next the work generalizes the evaluation to random initial portfolio selection, a technique frequently used in stock market studies. Although in practice portfolios are not built randomly, a random portfolio can be considered to represent a particular case of portfolio built by the conscious action of a broker. To build the random portfolios, the work first builds a reference portfolio containing 5 adwords chosen randomly from all topics collected in the three Twitter datasets. In order to ensure that this portfolio is feasible (*i.e.* it can satisfy the customer's constraint), the work checks if the sum of the number of clicks in the portfolio can eventually reach the target demand per day. If the random portfolio is not feasible, the work adds one other randomly chosen adword till the resulting portfolio becomes feasible. This results in the "*reference portfolio*" called portfolio A. Next, two "*augmented portfolios*" are built. The first augmented portfolio B is generated by adding randomly chosen adwords coming from trending and popular topics to the reference portfolio, while the second augmented portfolio C is generated by adding to the reference portfolio adwords coming only from the normal topics. The work limits the size of the augmented portfolio to 10 adwords that is the portfolio size suggested by Google [1].

The two augmented portfolios are used in order to compare the addition of trend-

ing and popular topics to the addition of the only normal topics with the same number of keywords in each portfolio. For each one of these three portfolios, the work derives the maximal ROI for a risk of 1 and compares the resulting ROIs. In order to decrease the impact of the randomness in the adwords choice, the work has generated independently 100 times the random reference portfolios along with the two attached augmented ones. The statistics over the 100 runs are analyzed finally.

Fig. 5-4 shows three cumulative distributions: the CDF of the $\overline{\overline{R(A)}}$, $\overline{\overline{R(B)}}$ and $\overline{\overline{R(C)}}$ obtained on each class of portfolio. It can be observed that the two augmented CDFs are clearly on the right side of the reference one, showing that augmenting the portfolio by both ways can improve the ROI. However, this curve does not tell which one of these augmentation ways is more profitable.

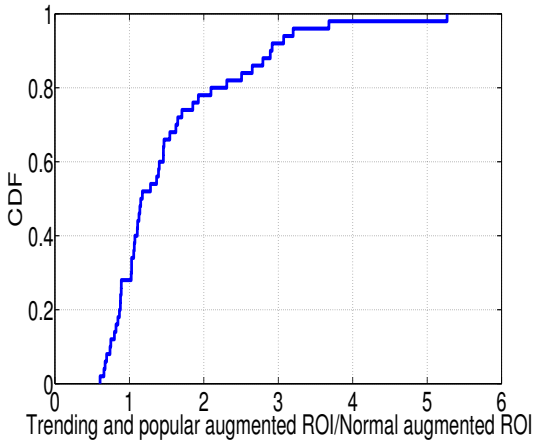


Figure 5-6: CDF of $\frac{\overline{\overline{R(B)}}}{\overline{\overline{R(C)}}}$

In order to determine which augmentation is more profitable, the work calculates the ratios of the ROI achieved by the two augmenting strategies to the ROI of the reference portfolio for a risk of 1. Fig. 5-5 shows the CDF of the two ratios $\frac{\overline{\overline{R(B)}}}{\overline{\overline{R(A)}}}$ and $\frac{\overline{\overline{R(C)}}}{\overline{\overline{R(A)}}}$ respectively. It can be observed that both augmenting methods achieve ratios that are always larger than 1, confirming that augmenting the portfolio always increases the achievable ROI. Moreover, the CDF curve for $\frac{\overline{\overline{R(B)}}}{\overline{\overline{R(A)}}}$ is on the right side of the CDF of $\frac{\overline{\overline{R(C)}}}{\overline{\overline{R(A)}}}$, meaning that the $\overline{\overline{R(B)}}$ is consistently larger than the $\overline{\overline{R(C)}}$. In particular, the average of $\frac{\overline{\overline{R(B)}}}{\overline{\overline{R(A)}}} = 5.20$, $\frac{\overline{\overline{R(C)}}}{\overline{\overline{R(A)}}} = 3.27$ and $\frac{\overline{\overline{R(B)}}}{\overline{\overline{R(C)}}} = 1.55$. Fig. 5-6 depicts

the CDF of $\frac{\overline{\overline{R(B)}}}{\overline{\overline{R(C)}}}$, which shows that in 28% of the cases the C portfolio achieves a better ROI while in the remaining 72% of the cases the portfolio B (augmented with popular and trending Twitter topics) has a better ROI.

5.7 Summary

Through an economic analysis of the third-party market, this study has developed a portfolio management framework that controls the tradeoff between the Return On Investment and the risk resulting from uncertainty on current CPC and achievable click number in a search engine marketing context. The work has studied the benefits of an efficient portfolio management, and in particular of the Efficient Frontier for comparing different portfolios. It has also proposed to use trending and popular topics extracted from Twitter to augment the adwords portfolios. The evaluation shows that the adwords augmentation is likely to improve the ROI on average by up to 4.2 times compared to a reference portfolio with the same level of risk.

This work opens ways for further researches investigating rational management of adwords portfolio. Even though this work considers the model's application from a broker's perspective, the results obtained here are also relevant for an advertiser acting himself as the broker for his own ads. Some limitations can be addressed as part of the future work. First, it is necessary to improve the evaluation experimental design by cooperating with a real broker. Second, finding relevant adwords amongst thousands of trending and popular topics may prove challenging. One possible approach consists of using ontologies that can characterize the semantic proximity of keywords. Such ontologies can be built through human expertise or automatically using Wikipedia [23]. This will facilitate the search for relevant trending and popular topics.

Chapter 6

Conclusion

This thesis makes an elaborate analysis of information diffusion in Microblogs. It firstly proposes an effective and unbiased sampling method which provides a basic and representative dataset for analysis and with the unbiased dataset, a Galton-Watson with Killing model is used to describe the information diffusion in Microblog. And then, the relationship between the information diffusion in Microblog and web interests is checked systematically which provides the reliable evidences that individual topics in Twitter and in the web share similar trending patterns both from the temporal and the spatial aspects while the trendiness in Twitter can precede for a few hours and is highly unstable compared to the one in web. Based on these observations, an economic analysis of the market involving a third-party ad broker is introduced and the potential of trending and popular topics coming from Twitter as adwords is discussed. The experiments show that the adwords augmenting strategy with the trending and popular topics in Twitter enables the broker to achieve, on average, four folds larger return on investment than with a non-augmented strategy, while still maintaining the same level of risk.

There are also some improvements needed in this work. At first, the GWK incorporates time as a discrete generational index, and does not account for the temporal dynamics of tweet diffusion. For this reason, a continuous time model that captures the temporal dynamics should be considered where the Markov birth-and-death process is a suitable candidate. Besides, in the work of adwords analysis the relevance

of Twitter inspired adwords for a given ad is not considered. Finding relevant adwords amongst thousands of trending and popular topics may prove very challenging and the practicability of the adwords augmenting strategy with Twitter trending and popular topics should be considered in the future.

Chapter 7

Publications

- Dong Wang, Mohamed-Ali Kaafar, Kavé Salamatian, and Gaogang Xie. Adwords Management for Third-parties in SEM: an Optimisation Model and the Potential of Twitter.(Accepted by INFOCOM 2016).
- Dong Wang, Zhenyu Li, and Gaogang Xie. Unbiased Sampling of Online Social Media with Local Disassortativity. (Submitted to IEEE Transactions on Knowledge and Data Engineering).
- Dong Wang and Gaogang Xie. Learning Trendiness from Twitter to Web: a Comparative Analysis of Microblog and Web Trending Topics. High Technology Letters transaction, 2016.
- Dong Wang, Hosung Park, Gaogang Xie, Sue Moon, Mohamed-Ali Kaafar, and Kavé Salamatian. A Genealogy of Information Spreading on Microblogs: a Galton-Watson-based Explicative Model. IEEE International Conference on Computer Communications (INFOCOM), 2013.
- Dong Wang, Mohamed-Ali Kaafar, Kavé Salamatian, and Gaogang Xie.What's trendy right now? A comparative analysis of Web and Microblogs trending topics.The 1st international conference on Internet Science, 2013.
- Jiali Lin, Zhenyu Li, Dong Wang, Kavé Salamatian, and Gaogang Xie.Analysis and Comparison of Interaction Patterns in Online Social Network and Social Media. IEEE International Conference on Computer Communications and Net-

works(ICCCN), 2012.

- Dong Wang, Zhenyu Li, Gaogang Xie, and Kavé Salamatian. The Pattern of Information Diffusion in Microblog, ACM International Conference on emerging Networking Experiments and Technologies (CoNEXT) student workshop, 2011.
- Dong Wang, Zhenyu Li, and Gaogang Xie. Towards Unbiased Sampling of Online Social Networks, IEEE International Conference on Communications (ICC), 2011.

Bibliography

[1] Adword setting. 2014. https://support.google.com/adwords/answer/2453981.

[2] Eytan Adar and Lada A. Adamic. Tracking information epidemics in blogspace. In *Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence*, 2005.

[3] Gagan Aggarwal, S Muthukrishnan, Dávid Pál, and Martin Pál. General auction mechanism for search advertising. In *Proceedings of the 18th WWW*. ACM, 2009.

[4] Yong-Yeol Ahn, Seungyeop Han, Haewoon Kwak, Sue Moon, and Hawoong Jeong. Analysis of topological characteristics of huge online social networking services. In *Proceedings of the 16th international conference on World Wide Web*, pages 835–844. ACM, 2007.

[5] Alexa. Alexa website. 2014. http://www.alexa.com.

[6] Jisun An, Meeyoung Cha, P Krishna Gummadi, and Jon Crowcroft. Media landscape in twitter: A world of new conventions and political diversity. In *Proceedings of the 5th ICWSM*. AAAI, 2011.

[7] Chris Anderson. The long tail. *Wired magazine*, 12(10):170–177, 2004.

[8] Konstantin Avrachenkov, Bruno Ribeiro, and Don Towsley. Improving random walk estimation accuracy with uniform restarts. In *Proceedings of the 7th Workshop on Algorithms and Models for the Web Graph*, 2010.

[9] Luca Becchetti, Carlos Castillo, Debora Donato, Adriano Fazzone, and I Rome. A comparison of sampling techniques for web graph characterization. In *Proceedings of the Workshop on Link Analysis (LinkKDD' 06), Philadelphia, PA*, 2006.

[10] Michael S Bernstein, Jaime Teevan, Susan Dumais, Daniel Liebling, and Eric Horvitz. Direct answers for search queries in the long tail. In *Proceedings of the SIGCHI*. ACM, 2012.

[11] Anand Bhalgat, Jon Feldman, and Vahab Mirrokni. Online allocation of display ads with smooth delivery. In *Proceedings of the 18th ACM SIGKDD*, pages 1213–1221. ACM, 2012.

[12] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[13] Johan Bollen, Huina Mao, and Xiaojun Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8, 2011.

[14] Béla Bollobás. *Modern graph theory*, volume 184. Springer, 1998.

[15] Anders Brodersen, Salvatore Scellato, and Mirjam Wattenhofer. Youtube around the world: geographic popularity of videos. In *Proceedings of the 21st international conference on World Wide Web*, pages 241–250. ACM, 2012.

[16] Anders Brodersen, Salvatore Scellato, and Mirjam Wattenhofer. Youtube around the world: Geographic popularity of videos. In *Proceedings of the 21st international conference on World Wide Web*, 2012.

[17] Juan Miguel Carrascosa, Roberto González, Rubén Cuevas, and Arturo Azcorra. Are trending topics useful for marketing?: visibility of trending topics vs traditional advertisement. In *Proceedings of the 1st COSN*. ACM, 2013.

[18] M. Cha, H. Haddadi, F. Benevenuto, and K.P. Gummadi. Measuring user influence in twitter: The million follower fallacy. In *Proceedings of the 4th International AAAI Conference on Weblogs and Social Media*, 2010.

[19] Meeyoung Cha, Haewoon Kwak, Pablo Rodriguez, Yongyeol Ahn, and Sue Moon. I tube, you tube, everybody tubes: Analyzing the worlds largest user generated content video system. In *Proceedings of the 7th ACM SIGCOMM Internet Measurement Conference*, 2007.

[20] Meeyoung Cha, Alan Mislove, Ben Adams, and Krishna P. Gummadi. Characterizing social cascades in flickr. In *Proceedings of the 1st Workshop on Online Social Networks*, 2008.

[21] Meeyoung Cha, Alan Mislove, and Krishna P. Gummadi. A measurement-driven analysis of information propagation in the flickr social network. In *Proceedings of the 18th international conference on World Wide Web*, 2009.

[22] Meeyoung Cha, Juan Perez, and Hamed Haddadi. Flash floods and ripples: The spread of media content through the blogosphere. In *Proceedings of the 3rd International AAAI Conference on Weblogs and Social Media*, 2009.

[23] Abdelberi Chaabane, Gergely Acs, and Mohamed A. Kaafar. You are what you like! Information leakage through users' Interests. In *Proceedings of the 19th Annual Network & Distributed System Security Symposium*, February 2012.

[24] Terence Chen, Abdelberi Chaabane, PierreUgo Tournoux, Mohamed-Ali Kaafar, and Roksana Boreli. How much is too much? leveraging ads audience estimation to evaluate public profile uniqueness. In *Proceedings of Privacy Enhancing Technologies (PETs)*, 2013.

[25] Ye Chen and Tak W Yan. Position-normalized click prediction in search advertising. In *Proceedings of the 18th ACM SIGKDD*, pages 795–803. ACM, 2012.

[26] Yifan Chen, Gui-Rong Xue, and Yong Yu. Advertising keyword suggestion based on concept hierarchy. In *Proceedings of the 2008 international conference on web search and data mining*, pages 251–260. ACM, 2008.

[27] Xu Cheng, Cameron Dale, and Jiangchuan Liu. Statistics and social network of youtube videos. In *Proceedings of the 16th International Workshop on Quality of Service*, 2008.

[28] A. Clauset, C. R. Shalizi, and M. E.J. Newman. Power-law distributions in empirical data. *SIAM Review 51(4)*, 2009.

[29] Challet Damien and Bel Hadj Ayed Ahmed. Predicting financial markets with google trends and not so random keywords. *arXiv preprint arXiv:1307.4643*, 2013.

[30] Anirban Dasgupta, Ravi Kumarand, and Tamas Sarlos. On estimating the average degree. In *Proceedings of the 23rd WWW*, 2014.

[31] Petros Venetis Eldar Sadikov, Aditya Parameswaran. Blogs as predictors of movie success. *Third International AAAI Conference on Weblogs and Social Media*, 2009.

[32] U. Frisch and D. Sornette. Extreme deviations and applications. *Journal of Physics I France*, 1997.

[33] Anindya Ghose and Sha Yang. An empirical analysis of search engine advertising: Sponsored search in electronic markets. *Management Science*, 55(10):1605–1622, 2009.

[34] Minas Gjoka, Maciej Kurant, Carter T Butts, and Athina Markopoulou. Walking in facebook: A case study of unbiased sampling of osns. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9. IEEE, 2010.

[35] Minas Gjoka, Maciej Kurant, Carter T Butts, and Athina Markopoulou. Practical recommendations on crawling online social networks. *IEEE Journal on Selected Areas in Communications*, 29(9):1872–1892, 2011.

[36] Gagan Goel and Aranyak Mehta. Online budgeted matching in random input models with applications to adwords. In *Proceedings of the 9th SODA*. ACM-SIAM, 2008.

[37] Michaela Goetz, Jure Leskovec, Mary McGlohon, and Christos Faloutsos. Modeling blog dynamics. In *Proceedings of the 3rd International AAAI Conference on Weblogs and Social Media*, 2009.

[38] M. Gomez-Rodriguez, J. Leskovec, and A. Krause. Inferring networks of diffusion and influence. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2010.

[39] Google. Keywordplanner. https://adwords.google.com/ko/KeywordPlanner/.

[40] Google. Google trends. 2012. http://www.google.com/trends.

[41] Google. Google stop words list. 2014. http://code.google.com/p/stop-words.

[42] Google. Quality score. 2014. https://support.google.com/adwords/answer/-2454010.

[43] D. Gruhl, R. Guha, D. Liben-Nowell, and A. Tomkins. Information diffusion through blogspace. In *Proceedings of the 13th international conference on World Wide Web*, 2004.

[44] Lei Guo, Enhua Tan, Songqing Chen, Zhen Xiao, and Xiaodong Zhang. The stretched exponential distribution of internet media access patterns. In *Proceedings of the 27th ACM Symposium on Principles of Distributed Computing*, 2008.

[45] Monika R Henzinger, Allan Heydon, Michael Mitzenmacher, and Marc Najork. On near-uniform url sampling. *Computer Networks*, 33(1):295–308, 2000.

[46] Price Water House Coopers IAB. Internet advertising revenue report 2012 full year results, 2013.

[47] Meredith Ringel Morris Jaime Teevan, Daniel Ramage. Twittersearch: a comparison of microblog search and web search. *In: Proceedings of the fourth ACM international conference on Web search and data mining*, pages 35–44, 2011.

[48] Amruta Joshi and Rajeev Motwani. Keyword generation for search engine advertising. In *Data Mining Workshops, 2006. ICDM Workshops 2006. Sixth IEEE International Conference on*, pages 490–496. IEEE, 2006.

[49] Sanjay R. Kairam, Meredith Ringel Morris, Jaime Teevan, Dan Liebling, and Susan Dumais. Towards supporting search over trending events with social media. *Seventh International AAAI Conference on Weblogs and Social Media*, 2013.

[50] R. Kannan, L. Lovász, and R. Montenegro. Blocking conductance and mixing in random walks. *Combinatorics Probability & Computing*, 15(4):541–570, 2006.

[51] D. Kempe, J. Kleinberg, and E. Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003.

[52] Balachander Krishnamurthy, Phillipa Gill, and Martin Arlitt. A few chirps about twitter. In *Proceedings of the first workshop on Online social networks*, pages 19–24. ACM, 2008.

[53] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, pages 79–86, 1951.

[54] Maciej Kurant, Minas Gjoka, Carter T Butts, and Athina Markopoulou. Walking on a graph with a magnifying glass: stratified sampling via weighted random walks. In *Proceedings of the ACM SIGMETRICS*. ACM, 2011.

[55] Maciej Kurant, Athina Markopoulou, and Patrick Thiran. On the bias of bfs (breadth first search). In *Proceedings of the 22nd International Teletraffic Congress (ITC)*. IEEE, 2010.

[56] Maciej Kurant, Athina Markopoulou, and Patrick Thiran. Towards unbiased bfs sampling. *Selected Areas in Communications, IEEE Journal on*, 29(9):1799–1809, 2011.

[57] Haewoon Kwak, , Hyunwoo Chun, and Sue Moon. Fragile online relationship: a first look at unfollow dynamics in twitter. In *Proceedings of the 2011 annual conference on Human factors in computing systems*, 2011.

[58] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is twitter, a social network or a news media? In *Proceedings of the 19th international conference on World wide web*, pages 591–600. ACM, 2010.

[59] J. Laherrere and D. Sornette. Stretched exponential distributions in nature and economy: Fat tails with characteristic scales. *European Physical Journal B 2*, 2008.

[60] Jong Gun Lee, Sue Moon, and Kave Salamatian. An approach to model and predict the popularity of online contents with explanatory factors. In *Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence*, 2010.

[61] Sang Hoon Lee, Pan-Jun Kim, and Hawoong Jeong. Statistical properties of sampled networks. *Physical Review E*, 73(1):016102, 2006.

[62] Ronny Lempel and Shlomo Moran. Rank-stability and rank-similarity of link-based web ranking algorithms in authority-connected graphs. *Information Retrieval*, 8(2):245–264, 2005.

[63] Kristina Lerman and Rumi Ghosh. Information contagion: an empirical study of the spread of news on Digg and Twitter social networks. In *Proceedings of the 4th International AAAI Conference on Weblogs and Social Media*, 2010.

[64] Jure Leskovec, Mary McGlohon, Christos Faloutsos, Natalie Glance, and Matthew Hurst. Cascading behavior in large blog graphs patterns and a model. In *Proceedings of the 7th SIAM International Conference on Data Mining*, 2007.

[65] Jianguo Lu and Dingding Li. Bias correction in a small sample from big data. *IEEE Transactions on Knowledge and Data Engineering*, 25(11):2658–2663, 2013.

[66] Jens Malmros, Naoki Masuda, and Tom Britton. Random walks on directed networks: Inference and respondent-driven sampling. *arXiv preprint arXiv:1308.3600*, 2013.

[67] Harry Markowitz. Portfolio selection. *The journal of finance*, 7(1):77–91, 1952.

[68] Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani. Adwords and generalized online matching. *Journal of the ACM*, 54(5):22, 2007.

[69] Alan Mislove, Massimiliano Marcon, Krishna P Gummadi, Peter Druschel, and Bobby Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 29–42. ACM, 2007.

[70] Jeonghoon Mo and Jean Walrand. Fair end-to-end window-based congestion control. *IEEE/ACM Transactions on Networking (ToN)*, 8(5):556–567, 2000.

[71] Helen Susannah Moat, Chester Curme, Adam Avakian, Dror Y Kenett, H Eugene Stanley, and Tobias Preis. Quantifying wikipedia usage patterns before stock market moves. *Scientific reports*, 3, 2013.

[72] Seth Myers, Chenguang Zhu, and Jure Leskovec. Information diffusion and external influence in networks. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2012.

[73] Esa Nummelin. *General irreducible Markov chains and non-negative operators*, volume 83. Cambridge University Press, 2004.

[74] Hosung Park and Sue Moon. Sampling bias in user attribute estimation of osns. In *Proceedings of the 22nd WWW*, 2013.

[75] M Piraveenan, M Prokopenko, and AY Zomaya. Local assortativeness in scale-free networks. *EPL (Europhysics Letters)*, 84(2):28002, 2008.

[76] Tobias Preis, Helen Susannah Moat, and H Eugene Stanley. Quantifying trading behavior in financial markets using google trends. *Scientific reports*, 3, 2013.

[77] Amir Hassan Rasti, Mojtaba Torkjazi, Reza Rejaie, Nick Duffield, Walter Willinger, and Daniel Stutzbach. Respondent-driven sampling for characterizing unstructured overlays. In *INFOCOM 2009, IEEE*, pages 2701–2705. IEEE, 2009.

[78] W.J. Reed and B.D. Hughes. On the distribution of family names. *Physica A: Statistical Mechanics and its Applications*, 319:579–590, 2003.

[79] Bruno Ribeiro and Don Towsley. Estimating and sampling graphs with multidimensional random walks. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pages 390–403. ACM, 2010.

[80] Bruno Ribeiro, Pinghui Wang, Fabricio Murai, and Don Towsley. Sampling directed graphs with random walks. In *INFOCOM, 2012 Proceedings IEEE*, pages 1692–1700. IEEE, 2012.

[81] Bruno F Ribeiro and Don Towsley. On the estimation accuracy of degree distributions from graph sampling. In *CDC*, pages 5240–5247, 2012.

[82] Oliver J Rutz and Randolph E Bucklin. A model of individual keyword performance in paid search advertising. *SSRN eLibrary*, 26, 2007.

[83] Tauhid R.Zaman, Ralf Herbrich, Jurgen van Gael, and David Stern. Predicting information spreading in twitter. In *Proceedings of Neural Information Processing Systems*, 2010.

[84] Eldar Sadikov, Montserrat Medina, Jure Leskovec, and Hector Garcia-Molina. Correcting for missing data in information cascades. In *Proceedings of the 4th Annual ACM International Conference on Web Search and Data Mining*, 2011.

[85] Salvatore Scellato, Cecilia Mascolo, Mirco Musolesi, and Jon Crowcroft. Track globally, deliver locally: improving content delivery networks by tracking geographic social cascades. In *Proceedings of the 20th international conference on World wide web*, pages 457–466. ACM, 2011.

[86] Sina. Sina news. 2012. http://news.sina.com.cn/o/2012-05-16/022524421550.shtml.

[87] Yaron Singer. How to win friends and influence people, truthfully: influence maximization mechanisms for social networks. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 733–742. ACM, 2012.

[88] Bernd Skiera, Jochen Eckert, and Oliver Hinz. An analysis of the importance of the long tail in search engine marketing. *Electronic Commerce Research and Applications*, 9(6):488–494, 2010.

[89] D. Sornette, F. Deschâtres, T. Gilbert, and Y. Ageon. Endogenous versus exogenous shocks in complex networks: An empirical test using book sale rankings. *Phys. Rev. Lett.*, 93:228701, Nov 2004.

[90] Timm O Sprenger, Andranik Tumasjan, Philipp G Sandner, and Isabell M Welpe. Tweets and trades: The information content of stock microblogs. *European Financial Management*, 2013.

[91] SpyFu. Spyfu statistics. 2014. http://www.spyfu.com/.

[92] Ajay Sridharan, Yong Gao, Kui Wu, and James Nastos. Statistical behavior of embeddedness and communities of overlapping cliques in online social networks. In *INFOCOM, 2011 Proceedings IEEE*, pages 546–550. IEEE, 2011.

[93] Daniel Stutzbach, Reza Rejaie, Nick Duffield, Subhabrata Sen, and Walter Willinger. On unbiased sampling for unstructured peer-to-peer networks. *IEEE/ACM Transactions on Networking (TON)*, 17(2):377–390, 2009.

[94] Stamatina Thomaidou and Michalis Vazirgiannis. Multiword keyword recommendation system for online advertising. In *Advances in Social Networks Analysis and Mining (ASONAM), 2011 International Conference on*, pages 423–427. IEEE, 2011.

[95] Trendistic. Trendistic website. 2014. http://trendistic.com.

[96] Ming-Hsiang Tsou, Jiue-An Yang, Daniel Lusher, Su Han, Brian Spitzberg, Jean Mark Gawron, Dipak Gupta, and Li An. Mapping social activities and concepts with social media (twitter) and web search engines (yahoo and bing): a case study in 2012 us presidential election. *Cartography and Geographic Information Science*, 40(4):337–348, 2013.

[97] Twitter. Twitter help center. 2012. https://support.twitter.com.

[98] Twitter. Twitter blog. 2014. http://blog.twitter.com/2011/06/global-pulse.html.

[99] Twitter. Twitter website. 2014. http://www.twitter.com.

[100] VA Vatutin and AM Zubkov. Branching processes. i. *Journal of Mathematical Sciences*, 39(1):2431–2475, 1987.

[101] Bimal Viswanath, Alan Mislove, Meeyoung Cha, and Krishna P Gummadi. On the evolution of user interaction in facebook. In *Proceedings of the 2nd ACM workshop on Online social networks*, pages 37–42. ACM, 2009.

[102] Dong Wang, Zhenyu Li, and Gaogang Xie. Towards unbiased sampling of online social networks. In *Communications (ICC), 2011 IEEE International Conference on*, pages 1–5. IEEE, 2011.

[103] Taifeng Wang, Jiang Bian, Shusen Liu, Yuyu Zhang, and Tie-Yan Liu. Psychological advertising: exploring user psychology for click prediction in sponsored search. In *Proceedings of the 19th ACM SIGKDD*, pages 563–571. ACM, 2013.

[104] Tianyi Wang, Yang Chen, Zengbin Zhang, Peng Sun, Beixing Deng, and Xing Li. Unbiased sampling in directed social graph. *ACM SIGCOMM Computer Communication Review*, 41(4):401–402, 2011.

[105] Xingxing Wang, Shijie Lin, Dongying Kong, Liheng Xu, Qiang Yan, Siwei Lai, Liang Wu, Alvin Chin, Guibo Zhu, Heng Gao, et al. Click-through prediction for sponsored search advertising with hybrid models. In *Proceedings of the 18th ACM SIGKDD CUP workshop*. ACM, 2012.

[106] Zhi Wang, Lifeng Sun, Chuan Wu, and Shiqiang Yang. Guiding internet-scale video service deployment using microblog-based prediction. *IEEE Infocom Proceedings*, 131(5):2901 – 2905, 2012.

[107] Wikipedia. Twitter wikipedia. 2014. http://en.wikipedia.org/wiki/Twitter.

[108] Karl Aberer Wojciech Galuba, Dipanjan Chakraborty, Zoran Despotovic, and Wolfgang Kellerer. Outtweeting the twitterers - predicting information cascades in microblogs. In *Proceedings of the 3rd Workshop on Online Social Networks*, 2010.

[109] Yusheng Xie, Zhengzhang Chen, Ankit Agrawal, Alok Choudhary, and Lu Liu. Random walk-based graphical sampling in unbalanced heterogeneous bipartite social graphs. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages 1473–1476. ACM, 2013.

[110] Jaewon Yang and Jure Leskovec. Modeling information diffusion in implicit networks. In *Proceedings of the 10th IEEE International Conference on Data Mining*, 2010.

[111] Jaewon Yang and Jure Leskovec. Patterns of temporal variation in online media. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 177–186. ACM, 2011.

[112] Shaozhi Ye, Juan Lang, and Felix Wu. Crawling online social graphs. In *Web Conference (APWEB), 2010 12th International Asia-Pacific*, pages 236–242. IEEE, 2010.

[113] Shaozhi Ye and Felix Wu. Measuring message propagation and social influence on twitter.com. In *Proceedings of the 2nd International Conference on Social Informatics*, 2010.

[114] Shuai Yuan, Jun Wang, and Xiaoxue Zhao. Real-time bidding for online advertising: measurement and analysis. In *Proceedings of the Seventh International Workshop on Data Mining for Online Advertising*, page 3. ACM, 2013.

[115] Xue Zhang, Hauke Fuehres, and Peter A Gloor. Predicting stock market indicators through twitter "i hope it is not as bad as i fear". *Procedia-Social and Behavioral Sciences*, 26:55–62, 2011.

[116] George K. Zipf. *Human Behaviour and the Principle of Least-Effort*. 1949. Addison-Wesley, Cambridge MA.