Enhance micro-blogging recommendations of posts with an homophily-based graph

Quentin Grossetti[1,2]

Supervised by Cédric du Mouza[2],
Camelia Constantin[1] and Nicolas Travers[2]

[1]LIP6 - Université Pierre Marie Curie - Paris, France

[2]CEDRIC Laboratory - CNAM - Paris, France

BDA - Novembre 2017

Introduction

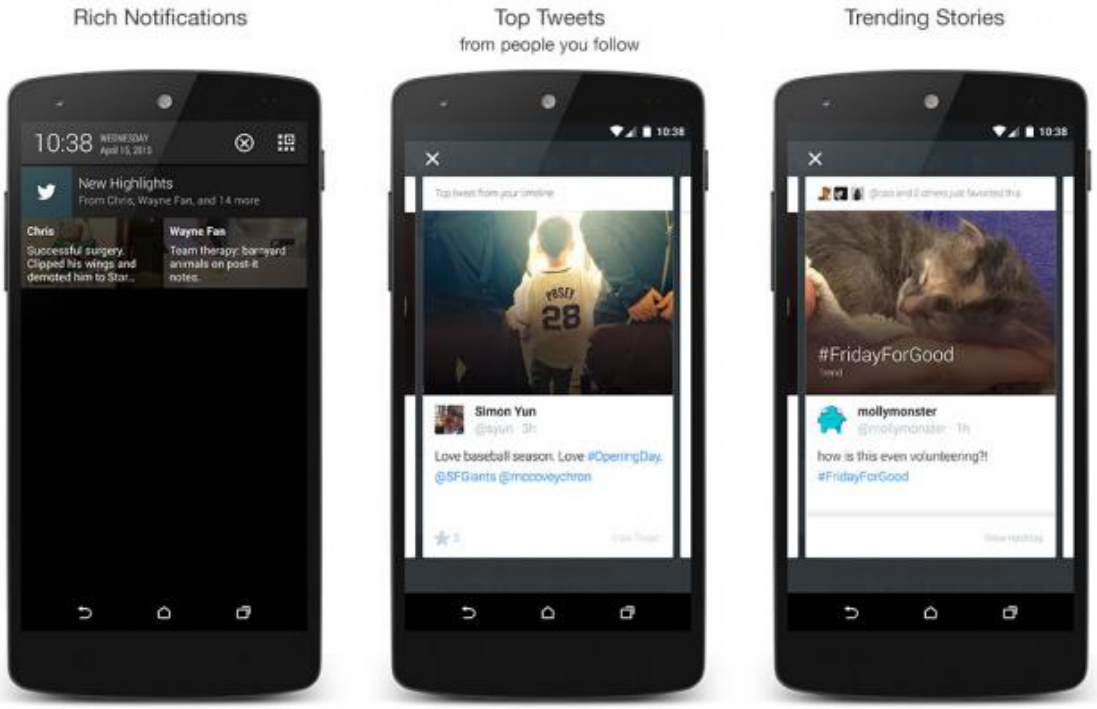
Context

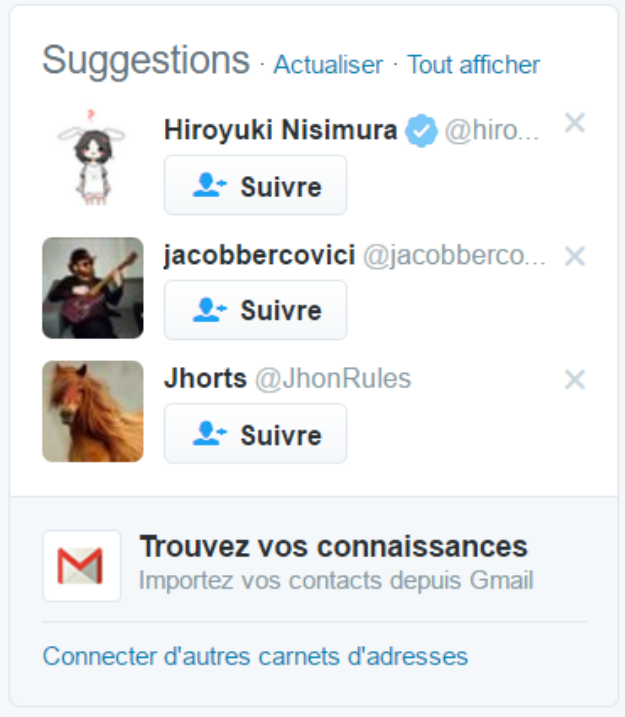
- Growth of microblogging plateforms since 2000

- 700 millions of messages/day in 2017
- 300 millions of messages/day in 2017
- 70 millions of publications/day in 2017
- 70 millions of pictures/day in 2017

Introduction

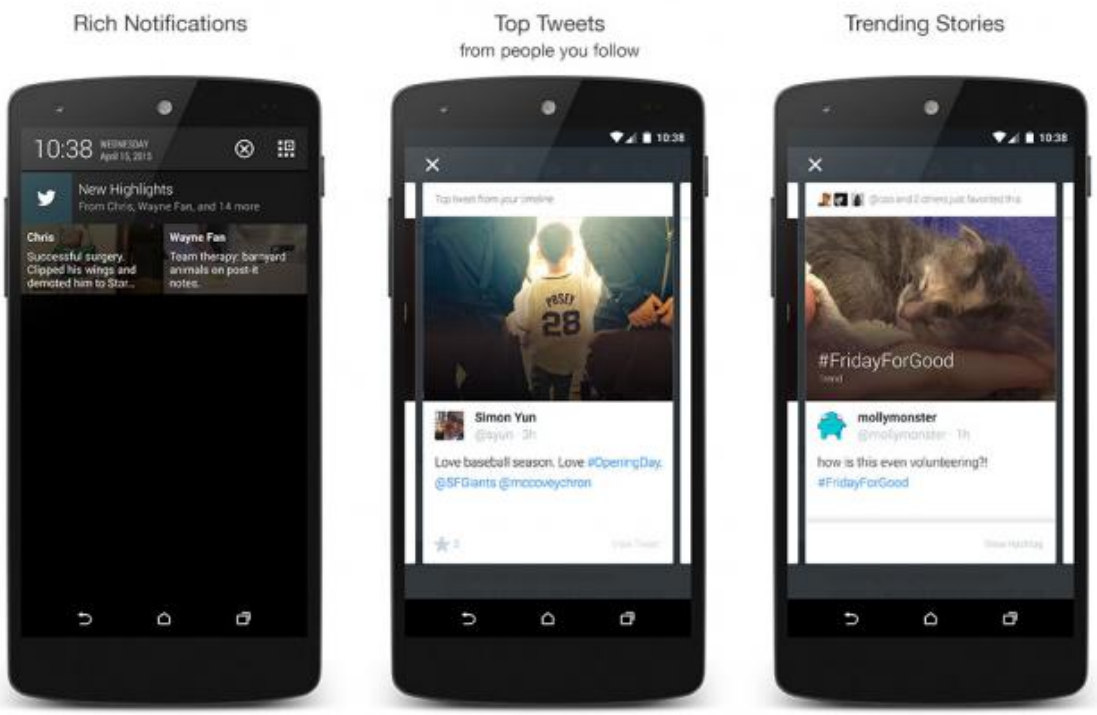
Real life examples

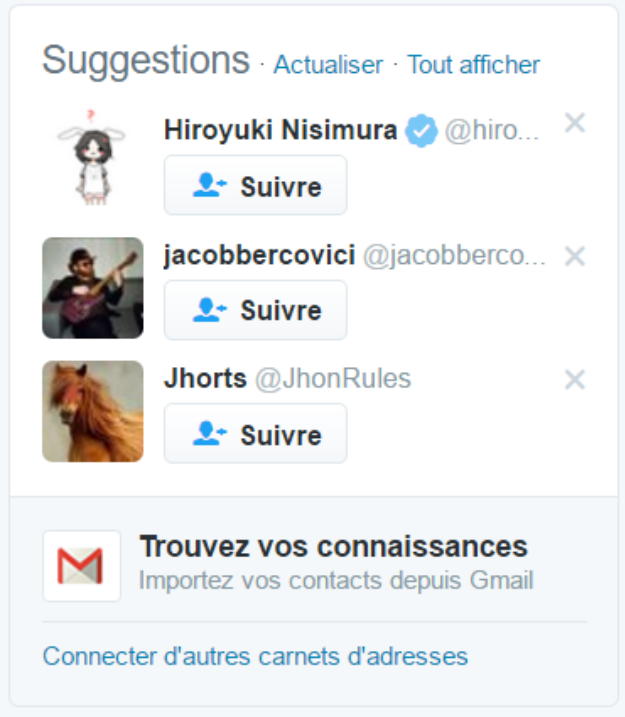




Introduction

Real life examples





Finding Users of Interest in Micro-blogging Systems (EDBT 2016)

Problem

How to connect users to relevant messages ?

- Recommendation of messages
- 700M new messages every day
- 300M of users
- Real time

Table of contents

State of the art

Content-based [Lops (2011)]

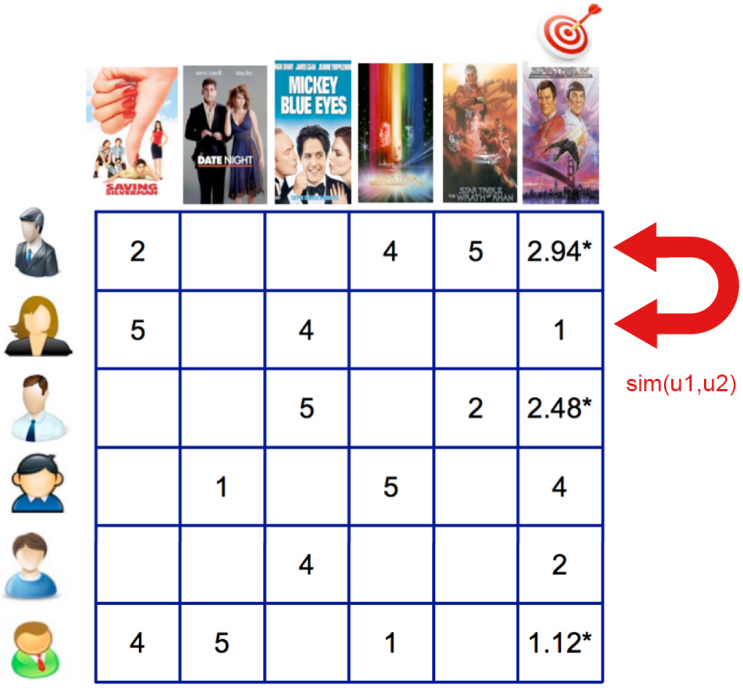
Extraction de caractéristiques

→ Réalisé par James Cameron
→ Avec DiCaprio et Kate Winslet
→ Romance, Drame
→ Le héros il meurt à la fin
→ Il y a un bateau dedans
→ Il y a du blanc sur l'affiche

Method	Pros	Cons
Content-based	No need of interactions	tweets are hard to describe

State of the art

Collaborative filtering [Schafer (2007)]



Method	Pros	Cons
Content-based	No need of interactions	tweets are hard to describe
Collaborative filtering	simple model and good results	too large matrix

State of the art

Matrix Factorization [Koren (2009)]

$$= \underbrace{\overset{U}{\begin{pmatrix} u_{11} & \cdots & u_{1r} \\ \vdots & \ddots & \\ u_{m1} & & u_{mr} \end{pmatrix}}}_{m \times r} \underbrace{\overset{S}{\begin{pmatrix} s_{11} & 0 & \cdots \\ 0 & \ddots & \\ \vdots & & s_{rr} \end{pmatrix}}}_{r \times r} \underbrace{\overset{V^{\mathsf{T}}}{\begin{pmatrix} v_{11} & \cdots & v_{1n} \\ \vdots & \ddots & \\ v_{r1} & & v_{rn} \end{pmatrix}}}_{r \times n}$$

Method	Pros	Cons
Content-based	No need of interactions	tweets are hard to describe
Collaborative filtering	simple model and good results	too large matrix
Matrix Factorization	efficient to fight sparsity	matrix growing too fast

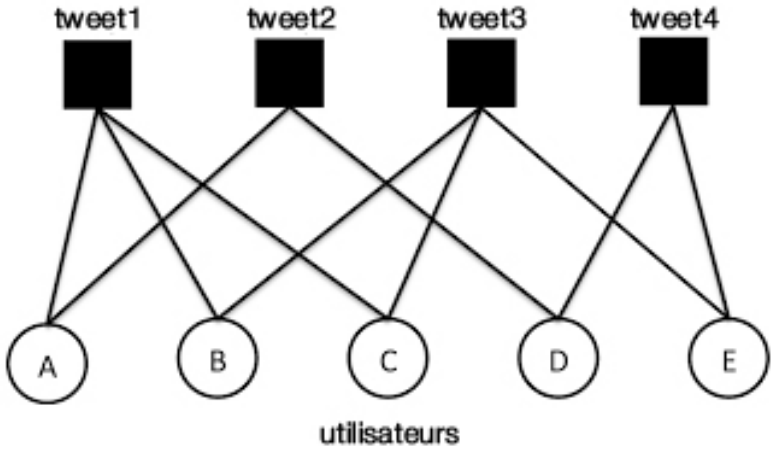
State of the art

Hybrid systems [Bostandjiev (2010)]

Method	Pros	Cons
Content-based	No need of interactions	tweets are hard to describe
Collaborative filtering	simple model and good results	too large matrix
Matrix Factorization	efficient to fight sparsity	matrix growing too fast
Hybrid systems	increase user engagement	hard to describe relationship

State of the art

Random walks models [Sharma (2016)]



Method	Pros	Cons
Content-based	No need of interactions	tweets are hard to describe
Collaborative filtering	simple model and good results	too large matrix
Matrix Factorization	efficient to fight sparsity	matrix growing too fast
Hybrid systems	increase user engagement	hard to describe relationship
Random walks models	very cheap	low memory

State of the art

Not only recommendations

- User recommendation (topology,content-based, demographic etc...)
- Hashtag (Bayesian model, euclidien...)
- Timeline Filtering (Deep Learning)
- Few papers on tweets recommendation except Twitter in 2016

Data Analysis

Dataset

Updated connected component from the graph found in [Kwak (2009)].

No of nodes	2,182,867
No of edges	325,451,980
No of tweets	2,571,173,369
Avg. out-degree	57.8
Avg. in-degree	69.4
max out-degree	348,595
max in-degree	185,401
Diameter	15
Average shortest path	3.7

TABLE – Twitter dataset characteristics

Data Analysis

Topology

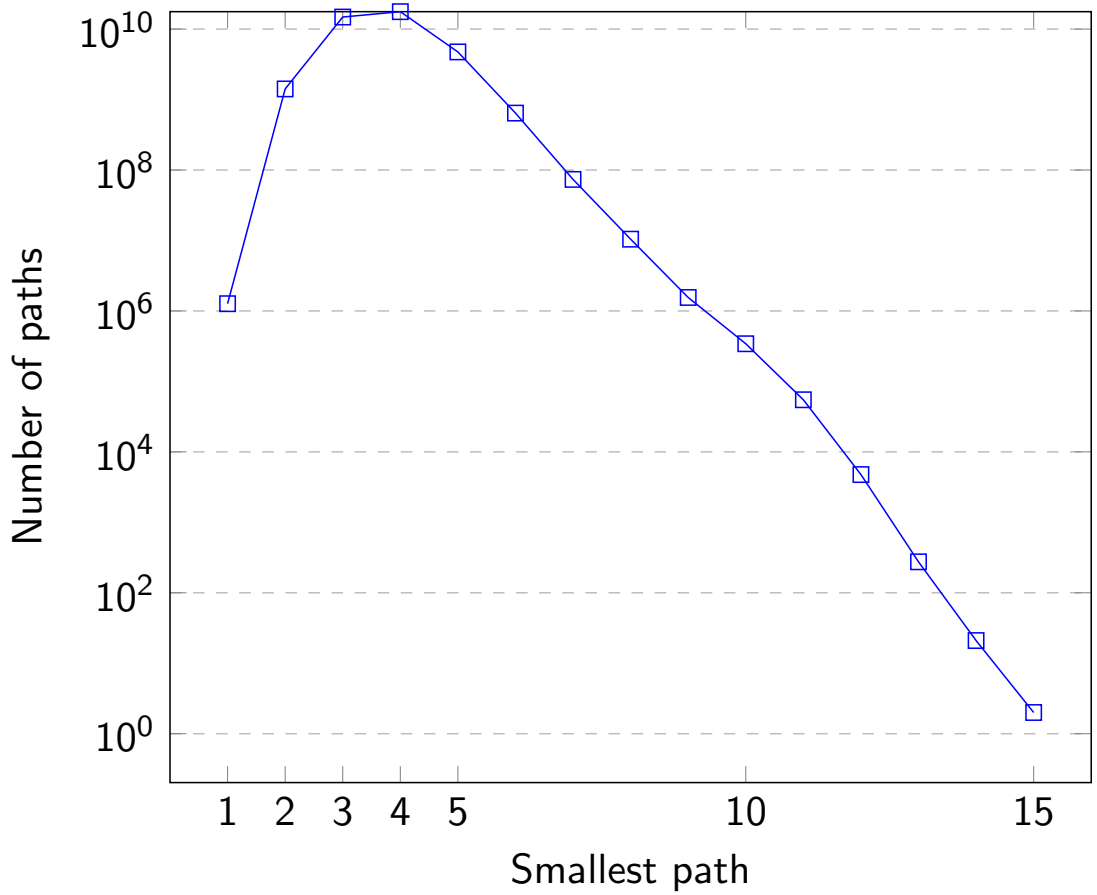
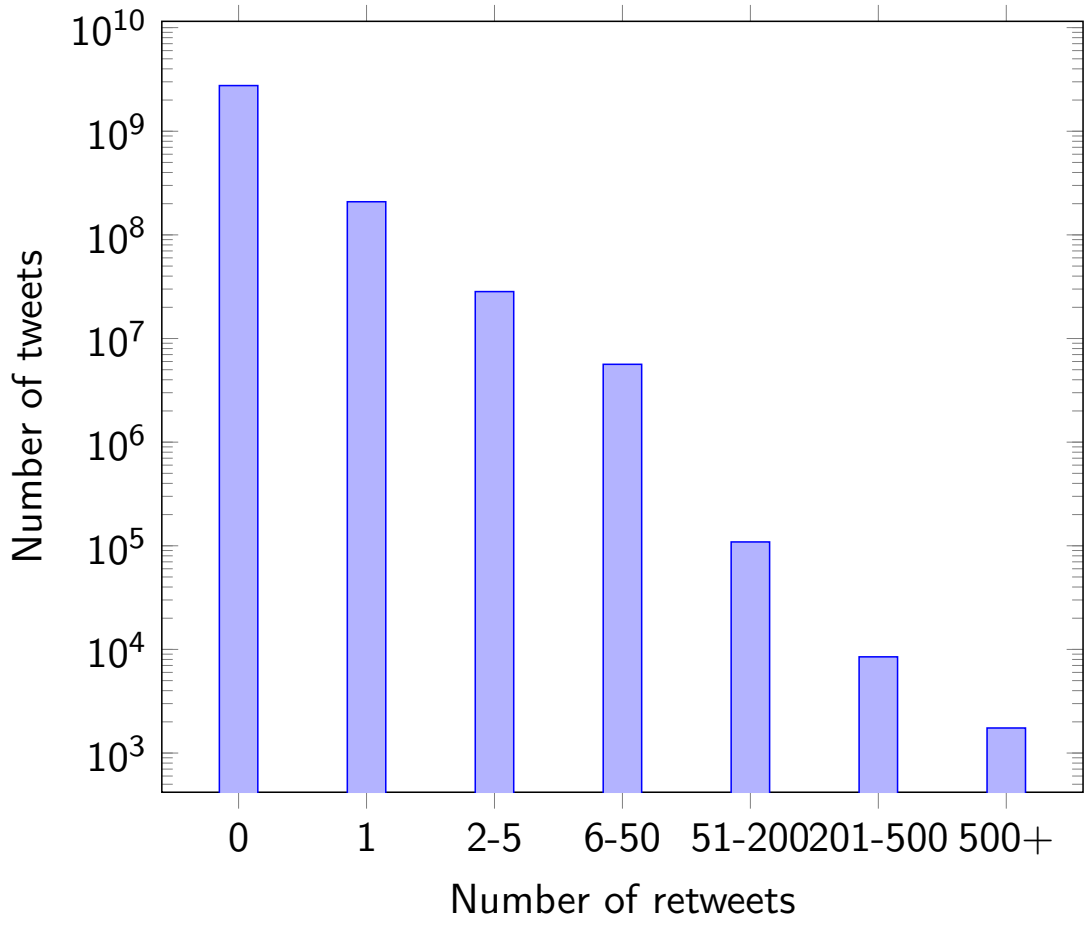


FIGURE – Twitter smallest paths distribution

- Small world with average distance of 3.7

Data Analysis

Retweets

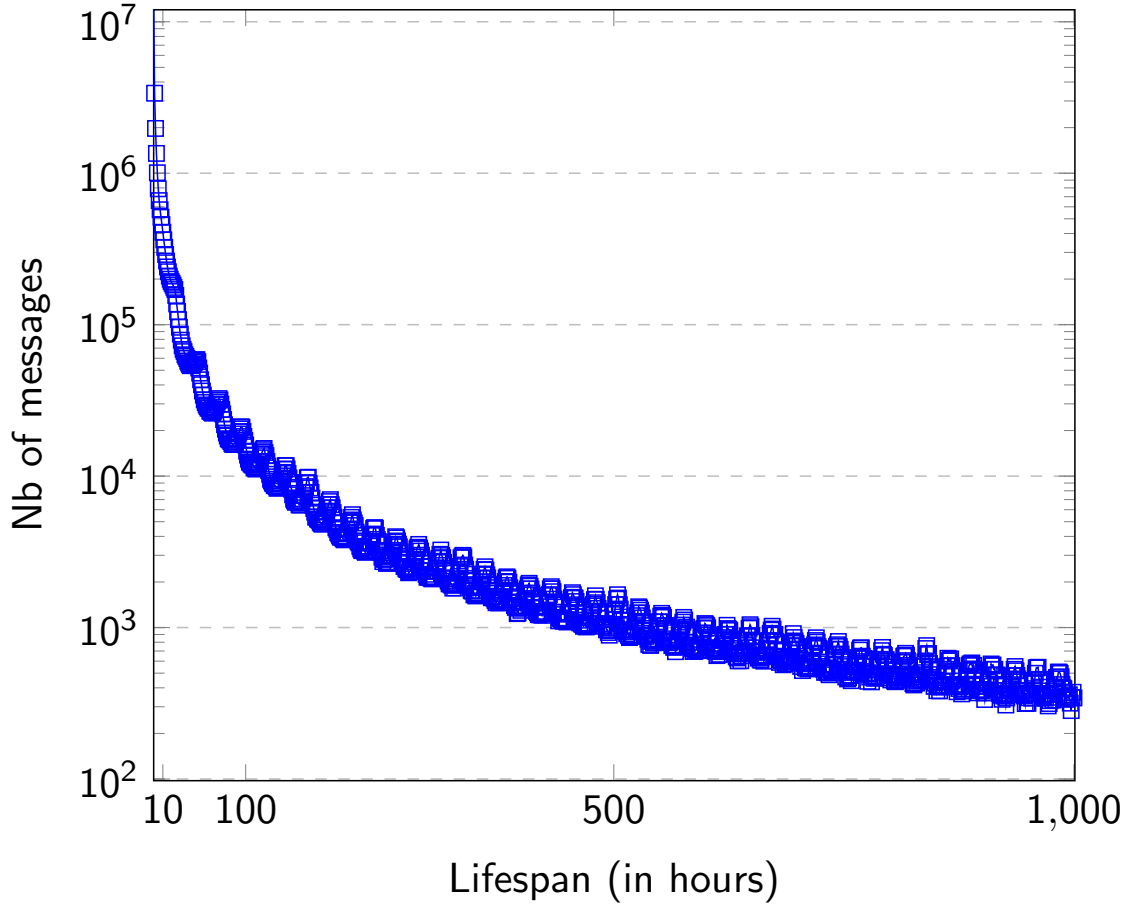


- 1 retweet - 7%
- 2-5 retweets - 1%
- 6+ - 0,2%

FIGURE – Distribution of the number of retweets per tweet

Data Analysis

Lifespan



- $< 1 hour$: 40%
- $< 3 days$: 90%

FIGURE – Lifespan of a message

Data Analysis

Homophily

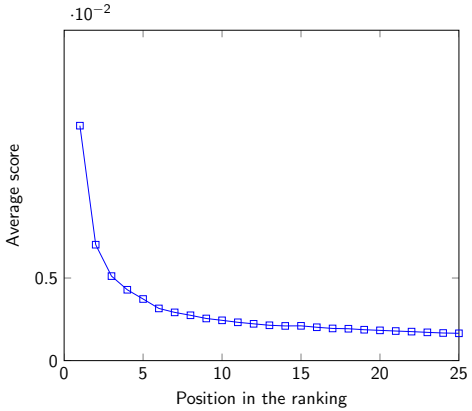
Distance	No of users	%	Mean similarity
1	3 229	02,65	0,0085
2	32 668	26,86	0,0014
3	81 645	67,13	0,0009
4	3 820	03,14	0,0010
5	43	00,03	0,0014
6	1	0	0,0008
Impossible	216	0,18	0,0017

TABLE – Evolution of the similarity score through distance in the network

$$sim(u,v) = \frac{\sum_{i \in L_u \cap L_v} \frac{1}{log(1+pop(i))}}{|L_u \cup L_v|} \quad (1)$$

Data Analysis

Homophily



		Distances distribution (%)			
Rank	Average Distance	1	2	3	4
1	1,55	57,03	31,53	10,64	0,8
2	1,68	49,60	33,13	16,87	0,4
3	1,8	42,45	36,02	20,72	0,8
4	1,86	38,71	38,71	20,56	2,02
5	1,98	31,44	40,16	27,59	0,81

TABLE – Link beetween distance in the network and position in the Top-N

Data Analysis

Conclusions

Many conclusions from this analysis :

- Freshness is crucial (Messages dies very fast)
  $\Rightarrow$ `real-time recommendation`
- Few users have high similarity
  $\Rightarrow$ `use transitivity`
- Distance 2 successfully gather important users
  $\Rightarrow$ `rely on this homophily`

Similarity Graph

Building process

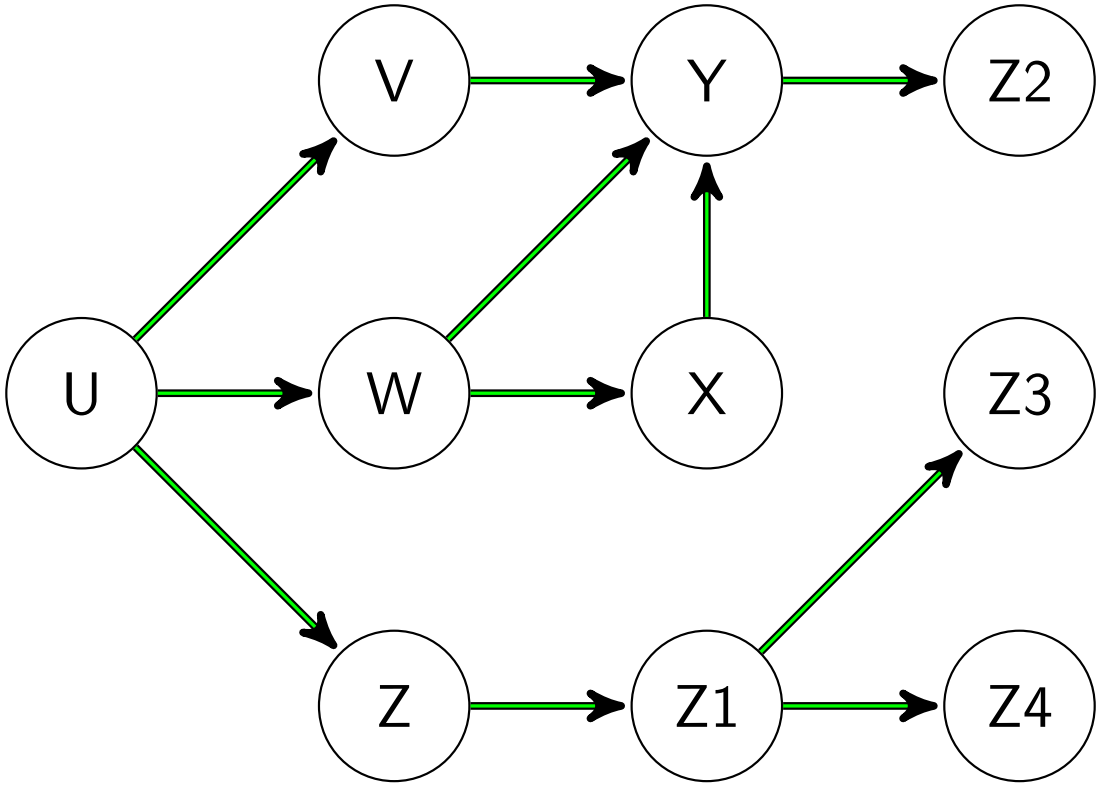


FIGURE – Twitter Graph

Graphe de similarité

Exemple de construction

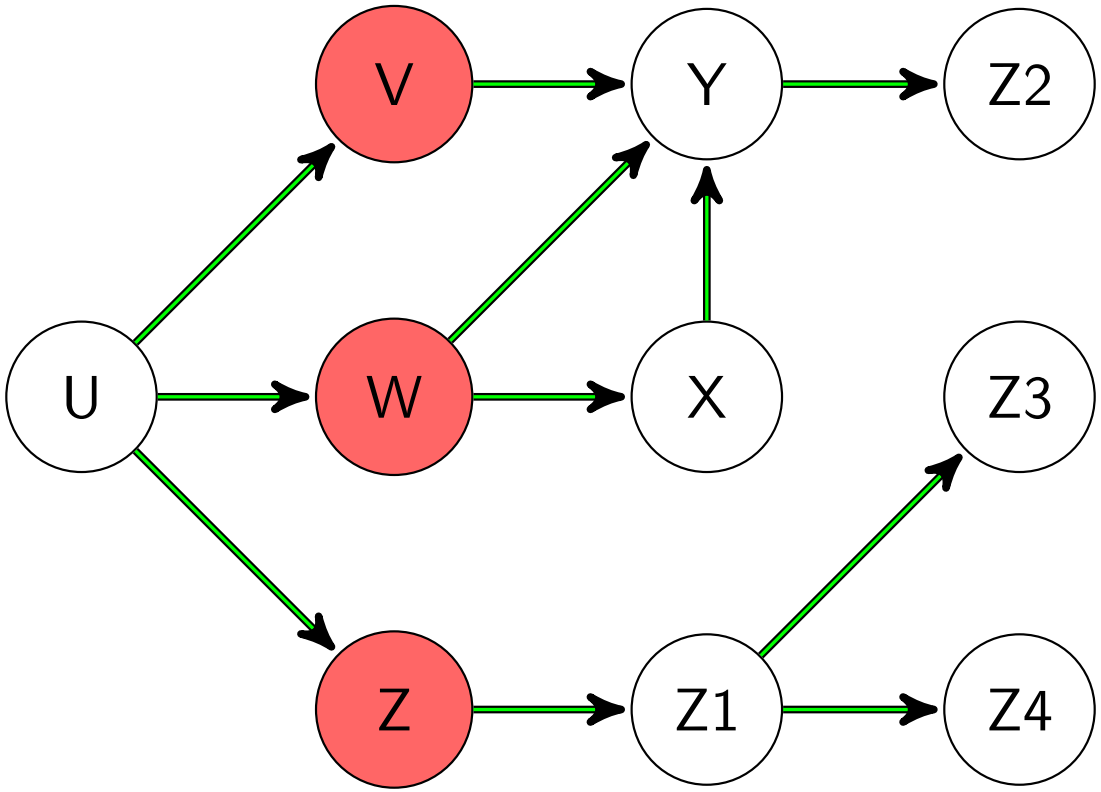


FIGURE – Twitter Graph

Similarity Graph

Building process

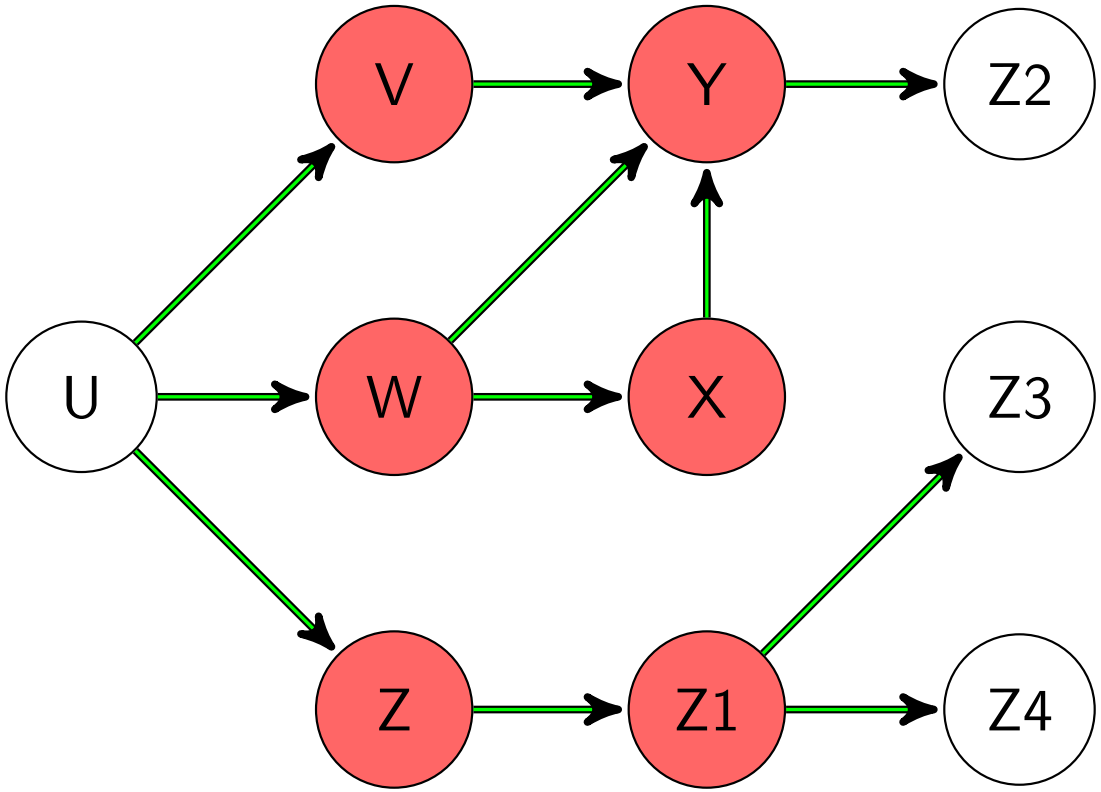


FIGURE – Twitter Graph

Graphe de similarité

Exemple de construction

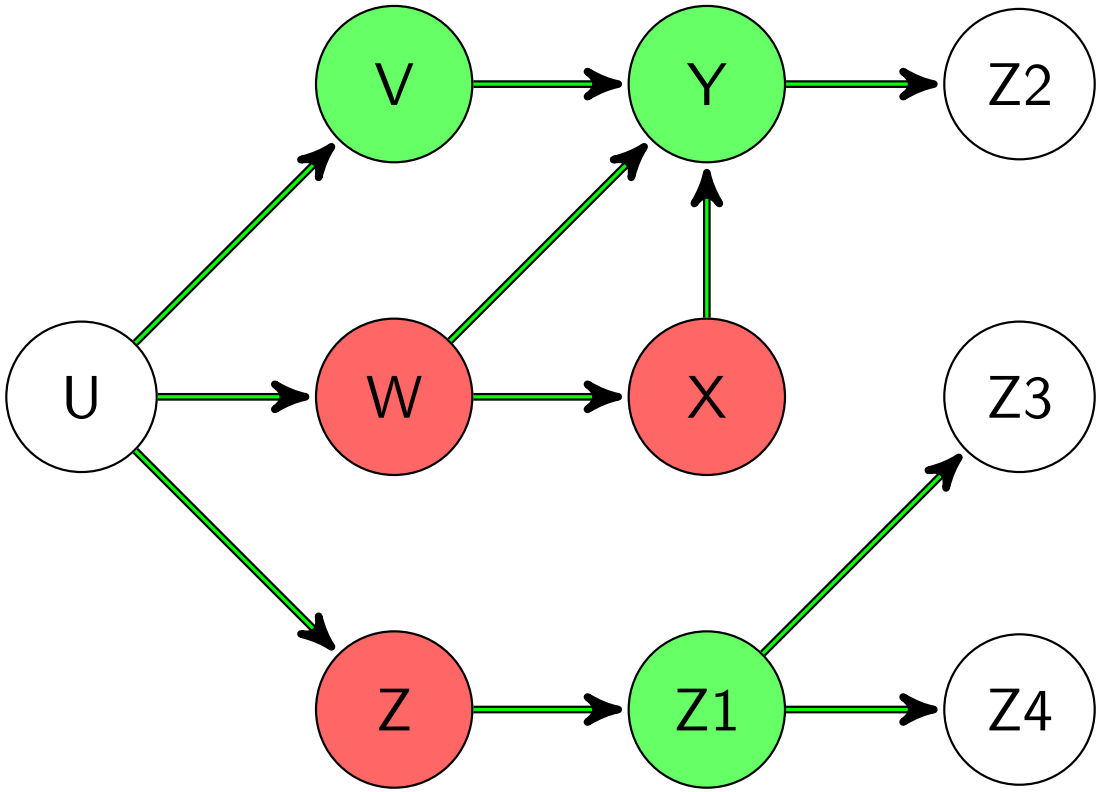


FIGURE – Twitter Graph

Similarity Graph

Building process

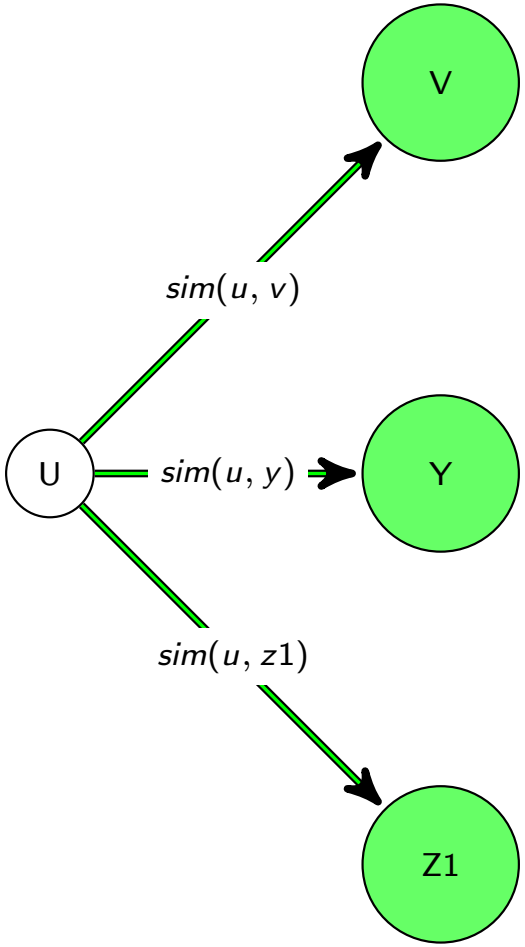


FIGURE – Similarity Graph

Similarity Graph

Characteristics

	Twitter Network	Similarity Graph
No of nodes	2 182 867	1 149 374
No of edges	325,451,980	4 950 417
Avg. similarity score		0.008
Mean out-degree	57.8	5.9

TABLE – Similarity Graph Characteristics

Propagation Model

In a nutshell

$$p(u,t) = \frac{\sum_{v \in Fu} p(u \leftarrow v, t)}{|Fu|} \tag{2}$$

With Fu the set of users influential to u and $p(u \leftarrow v, t)$ a probability estimation that u likes t determined by the behavior of the user v.

$$p(u \leftarrow v, t) = p(v,t) \times sim(u,v) \tag{3}$$

Propagation Model

Example

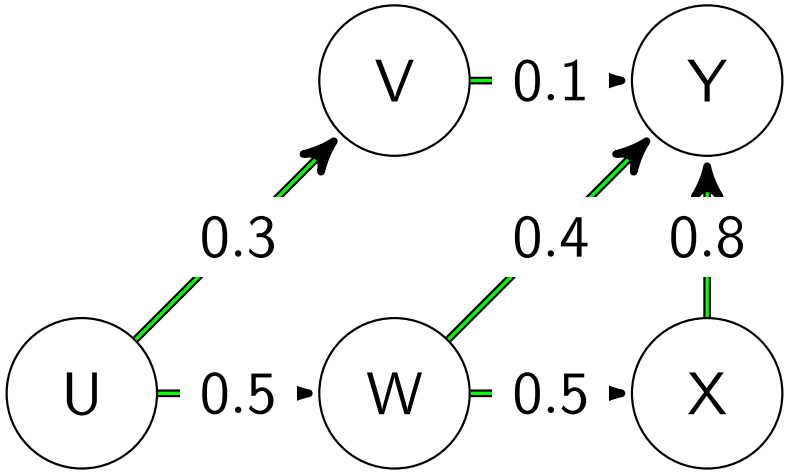


FIGURE – Propagation example

Propagation Model

Example

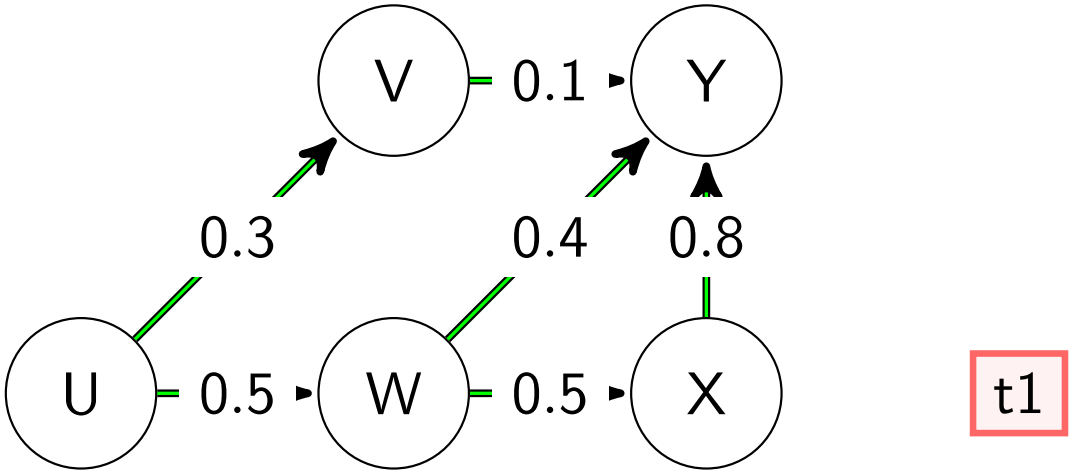


FIGURE – Propagation example - a tweet $t1$ is published

Propagation Model

Example

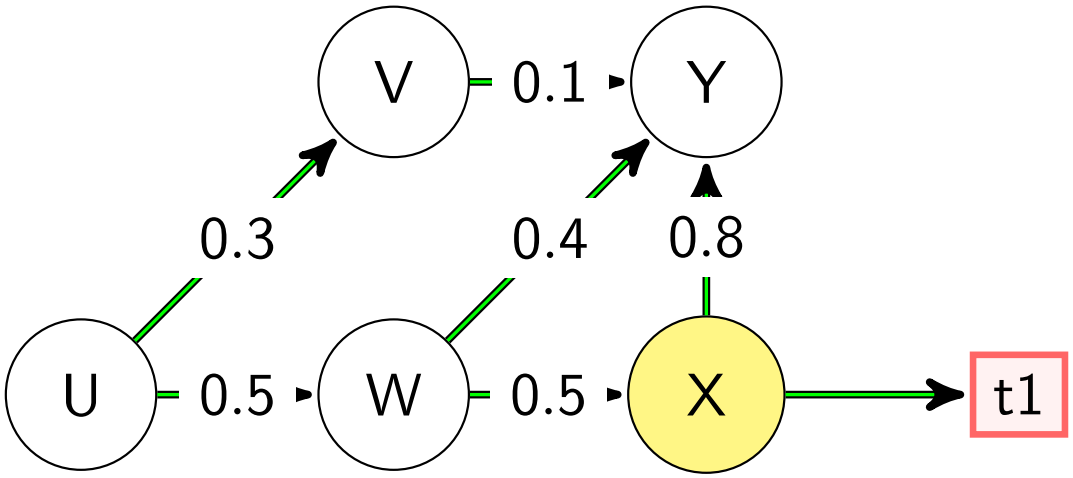


FIGURE – Propagation example - X shares/likes $t1$

$$p(x, t1) = 1$$

Propagation Model

Example

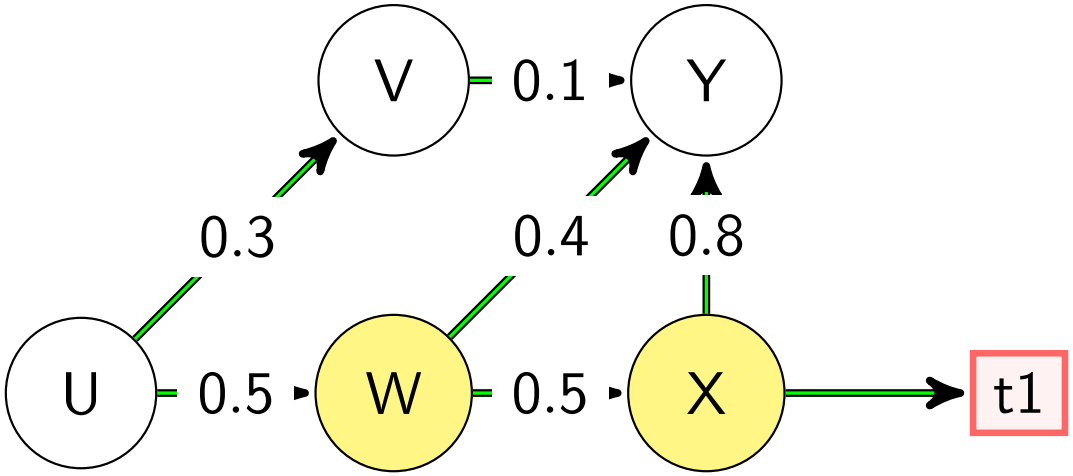


FIGURE – Propagation example - Propagation

$$p(w, t1) = \frac{\sum_{v \in Fw} p(w \leftarrow v, t)}{|Fw|} = \frac{0 + 1 \times 0.5}{2} = 0.25$$

Propagation Model

Example

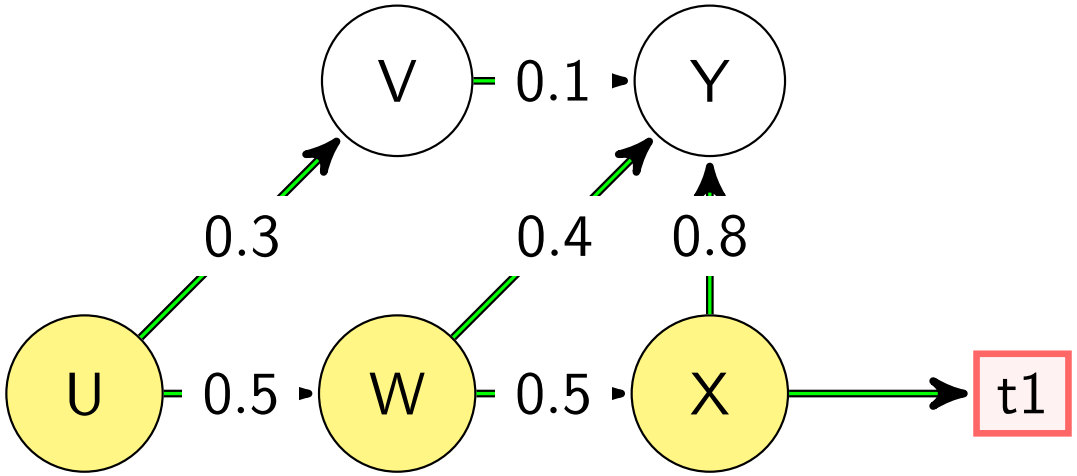


FIGURE – Propagation example - Propagation

$$p(u, t1) = \frac{0.25 \times 0.5}{2} = 0.0625$$

Propagation Model

Convergence

Let n be users $(u_1, u_2, ..., u_n)$:

$$\begin{cases} a_{11}p_{u_1} + a_{12}p_{u_2} + ... + a_{1n}p_{u_n} & = & b_1 \\ a_{21}p_{u_1} + a_{22}p_{u_2} + ... + a_{2n}p_{u_n} & = & b_2 \\ \dots & = & \dots \\ a_{n1}p_{u_1} + a_{n2}p_{u_2} + ... + a_{nn}p_{u_n} & = & b_n \end{cases}$$

Could also be written as $Ap = b$ with

$$A = \begin{array}{c} \\ u_1 \\ u_2 \\ \vdots \\ u_n \end{array} \begin{array}{c} \begin{array}{cccc} u_1 & u_2 & \cdots & u_n \end{array} \\ \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \end{array} p = \begin{pmatrix} p(u_1) \\ p(u_2) \\ \vdots \\ p(u_n) \end{pmatrix} b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

Because $\forall u, v \ sim(u, v) \leq 1$, $|a_{jj}| \geq \sum_{j \neq i} |a_{ij}|$ for every i, the matrix A is diagonally dominant.

Propagation Model

Optimizations

- Speed up the convergence
  Let $\Delta(u, t1) = p(u, t)^{k+1} - p(u, t)^k$
  If $\Delta(u, t1) < \beta$ we stop the propagation

Propagation Model

Optimizations

- `Speed up the convergence`
  Let $\Delta(u, t1) = p(u, t)^{k+1} - p(u, t)^k$
  If $\Delta(u, t1) < \beta$ we stop the propagation
- `Limitation of popular messages`
  If $p(u, t) < f(t)$ no need to propagate.
  $f(t) = 1 - \frac{k^p}{k^p + pop(t)^p}$

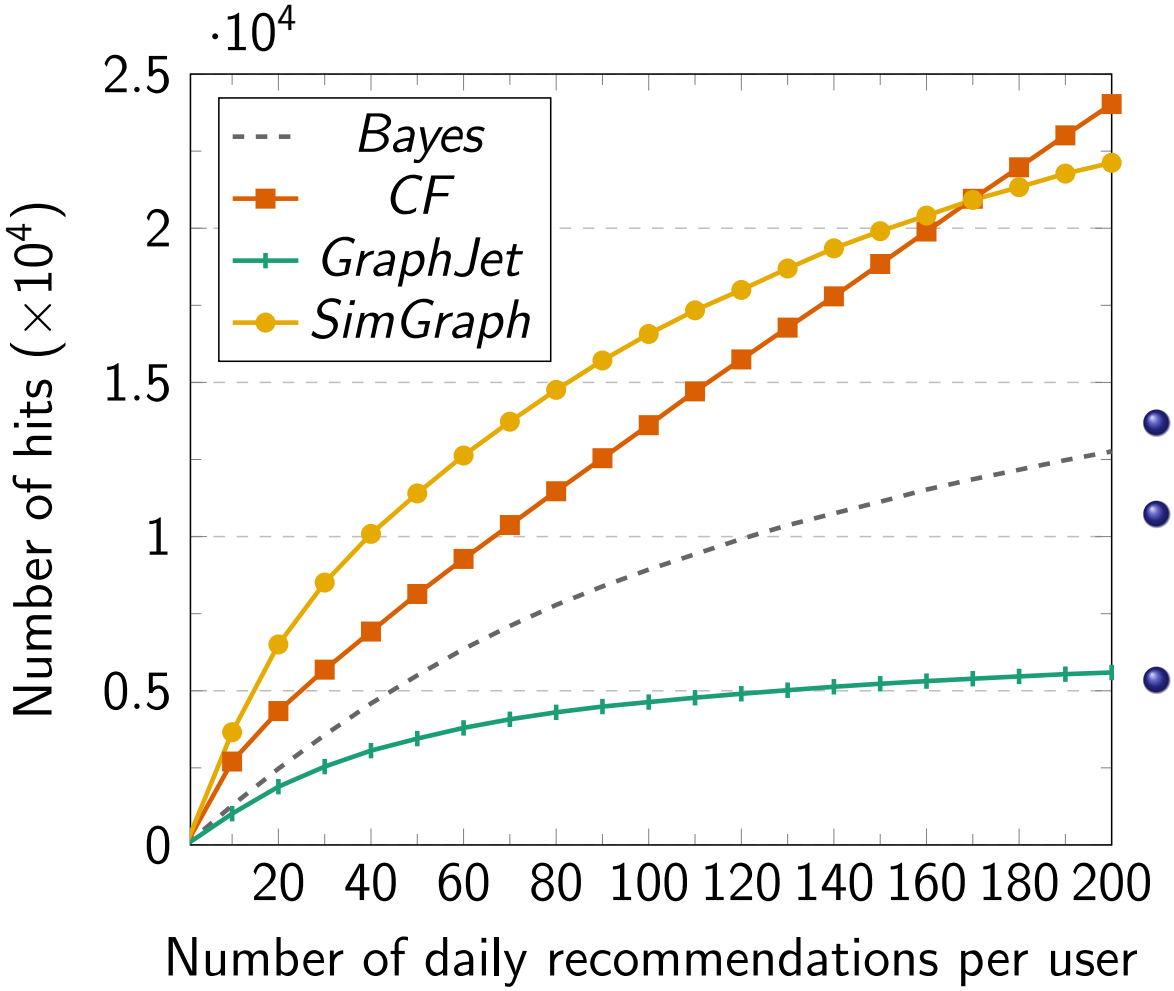
Experiments

Protocol

- 130 Millions of messages shared at least twice
- Split the ranked set 90% - 10%
- Compute recommendation during this 10% for 1500 random users (500 small, 500 medium, 500 big)
- Comparison with
  - `CF` : naive collaborative filtering
  - `Bayes` : probabilistic model
  - `GraphJet` : Twitter used solution

Experiments

Hits



- Linear growth of *CF*
- Fast growth for *SimGraph*
- *GraphJet* stuck around 5000 hits

FIGURE – Hits pour 1500 utilisateurs

Experiments

Hits according to user profiles

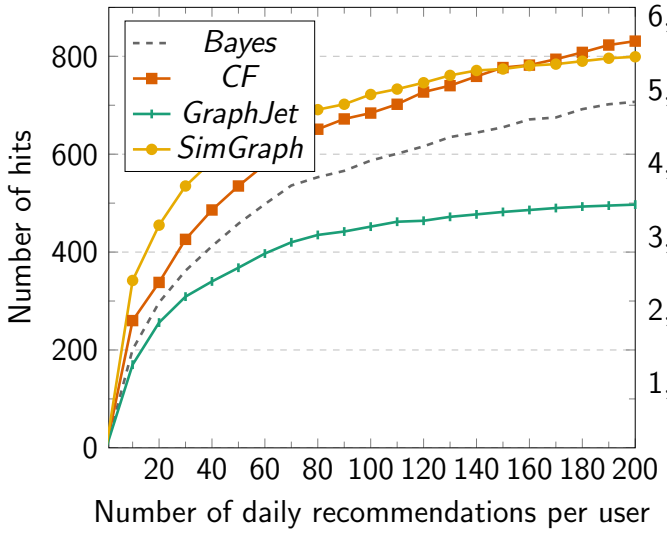


FIGURE – 500 *small*

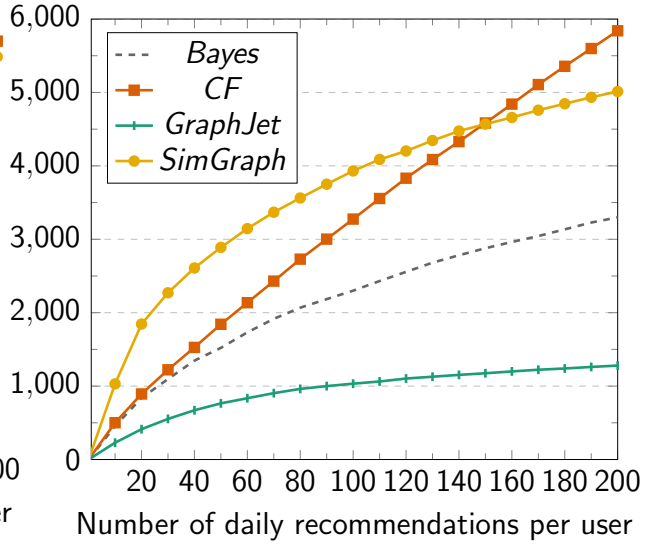


FIGURE – 500 *medium*

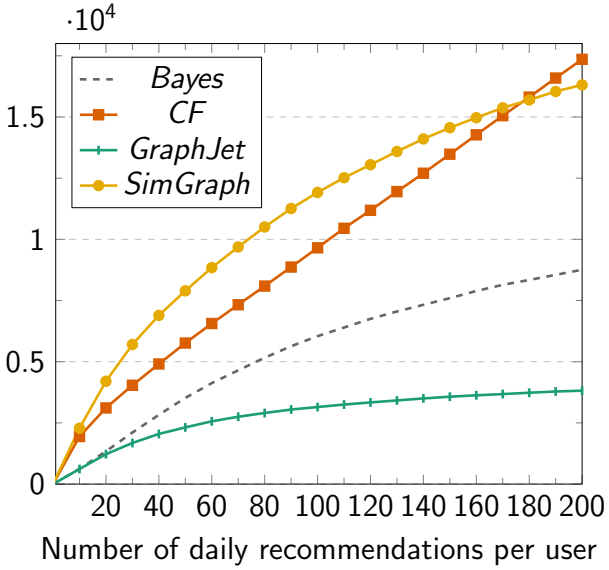


FIGURE – 500 *big users*

small < 50 ; medium < 1000 ; big > 1000

Tendencies are very stables no matter the profile of users

Experiments

Hits accuracy

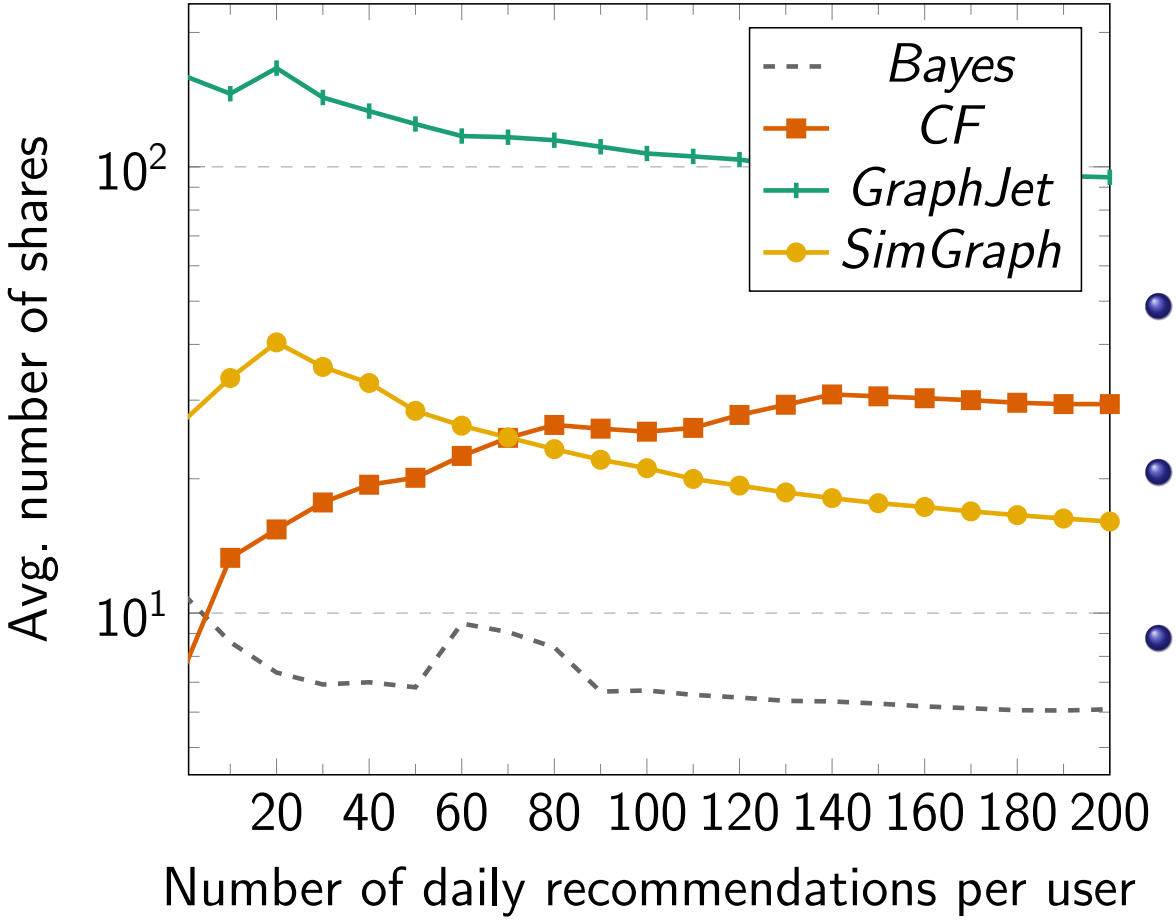
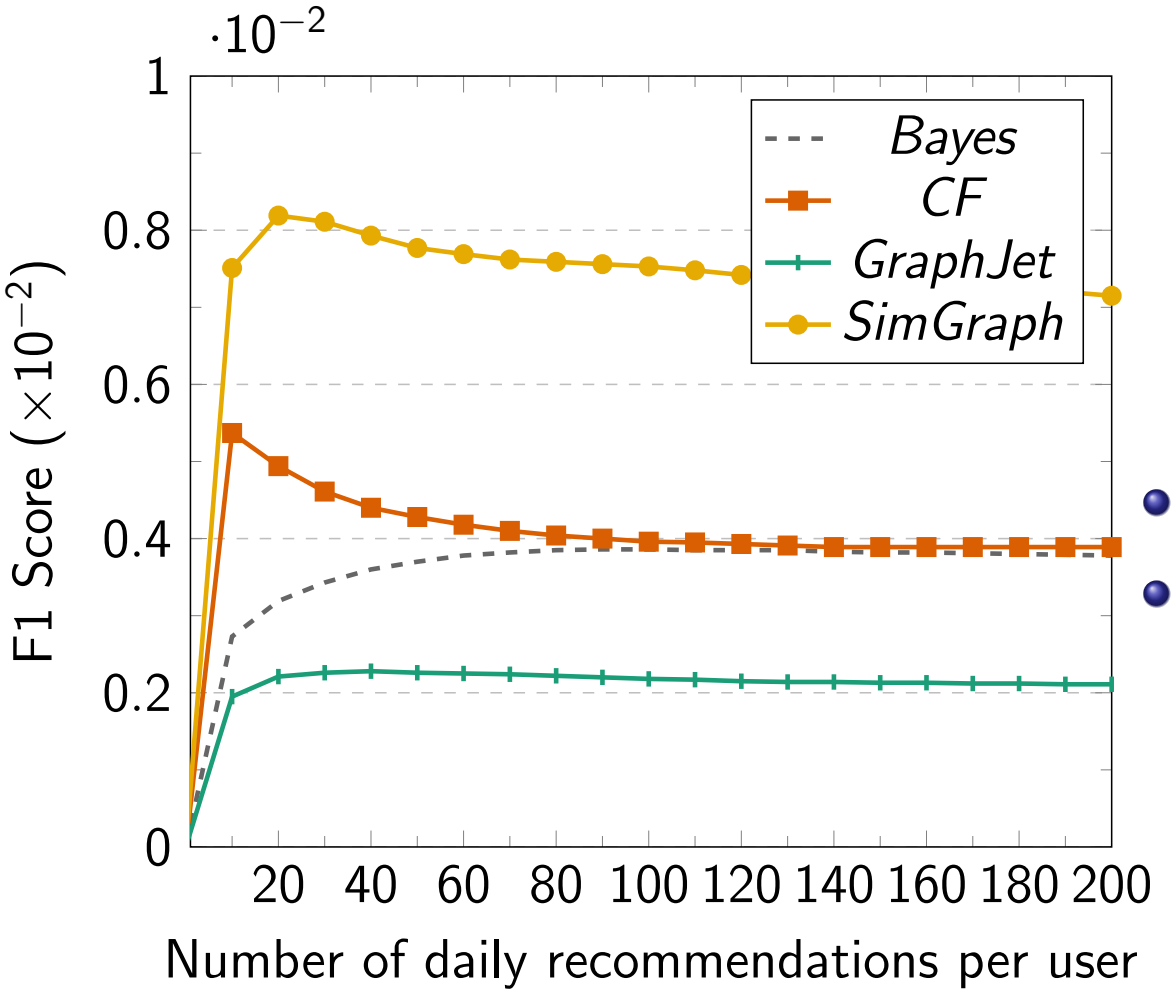


FIGURE – Hits popularity

- *Bayes* targets close messages
- *GraphJet* targets popular messages
- *CF* and *SimGraph* are mixing both popular and close messages

Experiments

F1 scores



- Small values
- Peak around 20 recommendations

FIGURE – F1 Scores

Experiments

Running time

	init. (per user)	**init total time** 1,149,374 users	**time (per message)**	**total time (70 cores //)** 13,238,941 Tweets (Trial period)	**total time** init + recos
Bayes	10ms	0.04h	975ms	51.22h	51.26h
CF	8,583ms	39.40h	0.5ms	0.02h	41.01h
SimGraph	311ms	1.41h	38ms	2.00h	3.41h
	init. (per user)	**init total time** 1,149,374 users	**time (per user)**	**total time (70 cores //)** 1,149,374 users * 66 days (Trial period)	**total time** init + recos
GraphJet	0ms	0h	14ms	4.2h	4.2h

TABLE – Initialization and recommendation time (in ms)

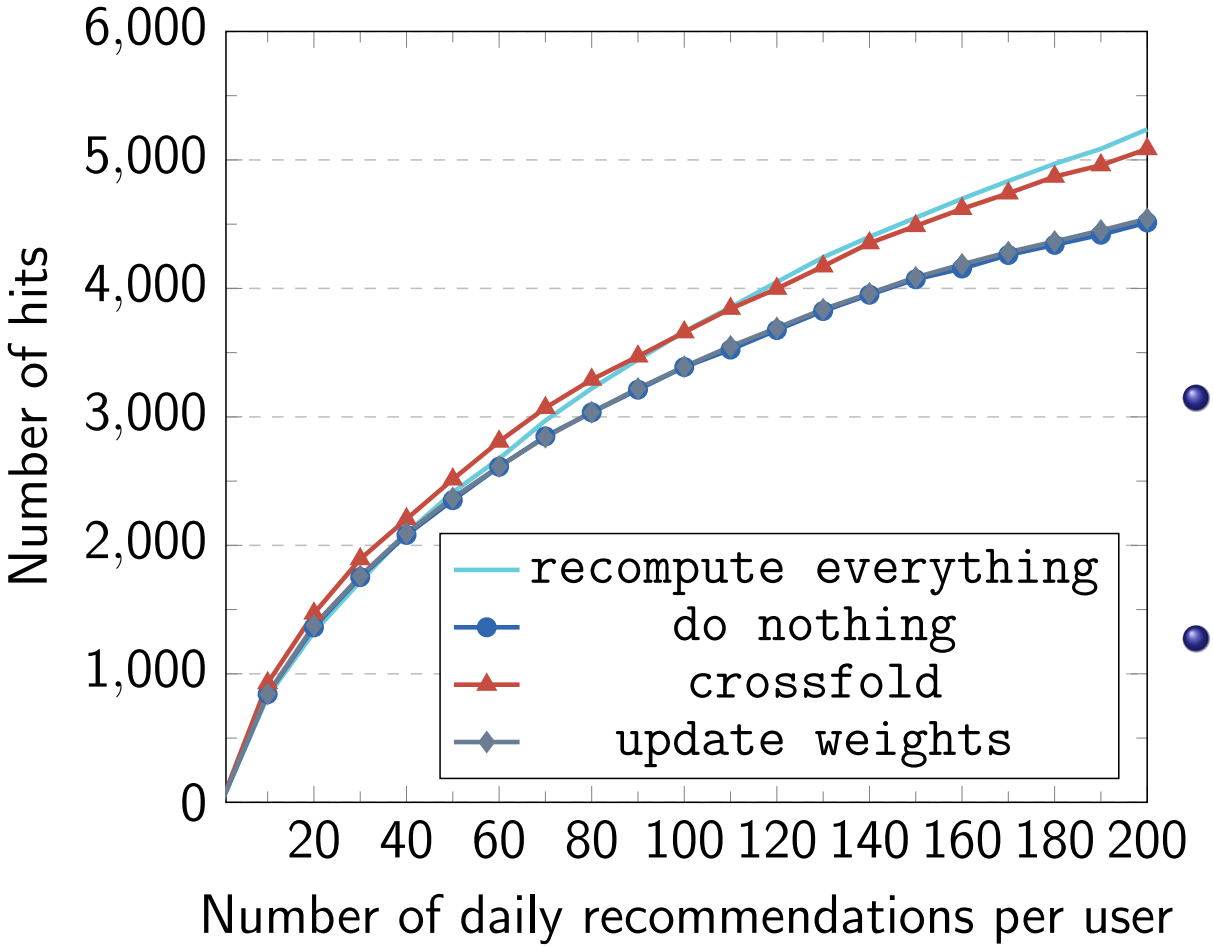
Experiments

Updating strategies

- How to update *SimGraph* ?
- Split the last 10% in 2
- Evaluate hits prediction impact for the remaining 5% :
  - *do nothing*
  - *recompute everything*
  - *update only weights*
  - *crossfold*

Experiments

Updating strategies



- doing nothing is the same as updating weights
- crossfold (very cheap) works very well

FIGURE – Hits / updating strategies

Experiments

Convergence property of the SimGraph

Iteration	Number of edges
1	4 950 417
2	7 519 031
3	10 836 129
4	11 496 445
5	11 678 747

TABLE – Number of edges evolution through iterations

Conclusion

Contribution

- Construction and analysis of a large Twitter dataset
- Method relying on homophily to find nearest neighbors at low cost
- Construction and optimization of a convergent propagation model
- Comparison of the recommendations made by our model with state of the art solutions
- Possibility for the model to be updated at low cost

Conclusion

Future works

- Densify points of comparison between users
- Burst recommendation bubbles
- Work on the *crossfold* convergence of the model
- Add a popularity prediction optimization

Thanks for you attention !

ANNEXES

Annexes

Lifespan and popularity

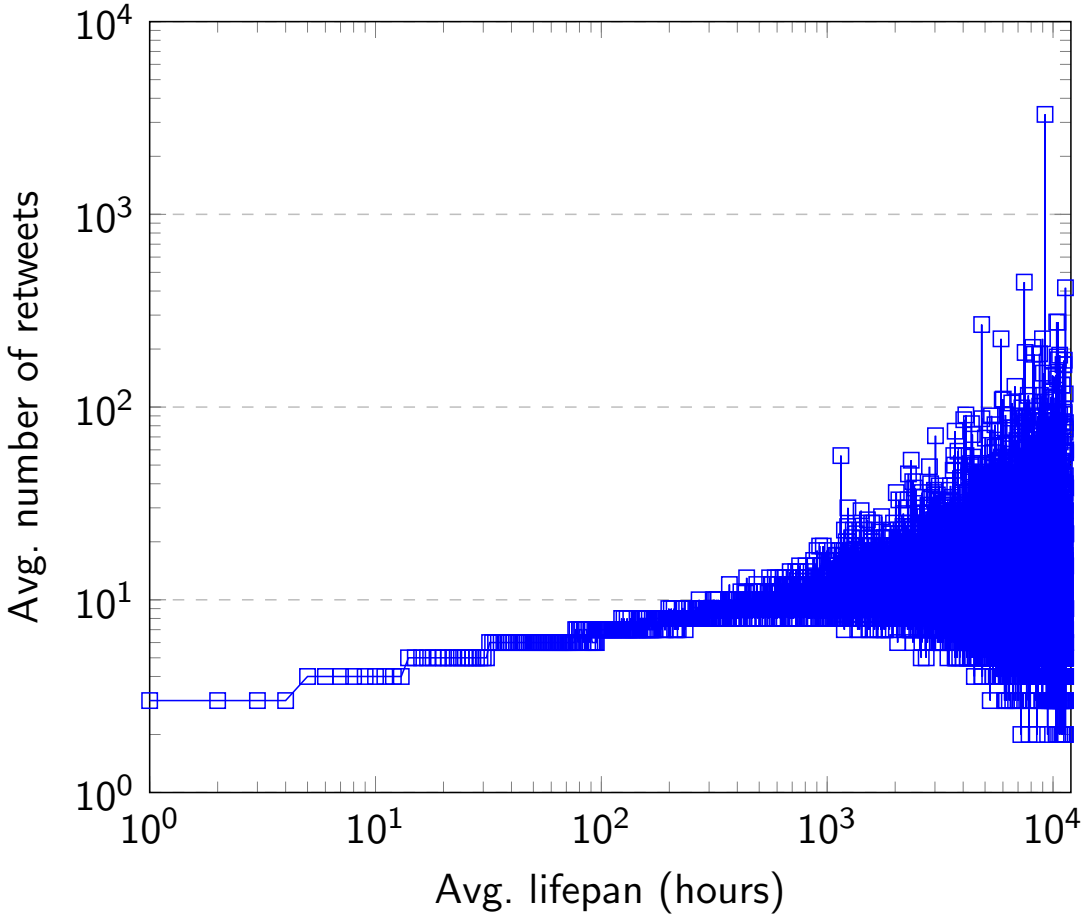


FIGURE – Correlation between lifespan and popularity

- Strong correlation up to 10^3 hours
- After a month, the correlation fades

Annexes

Topology

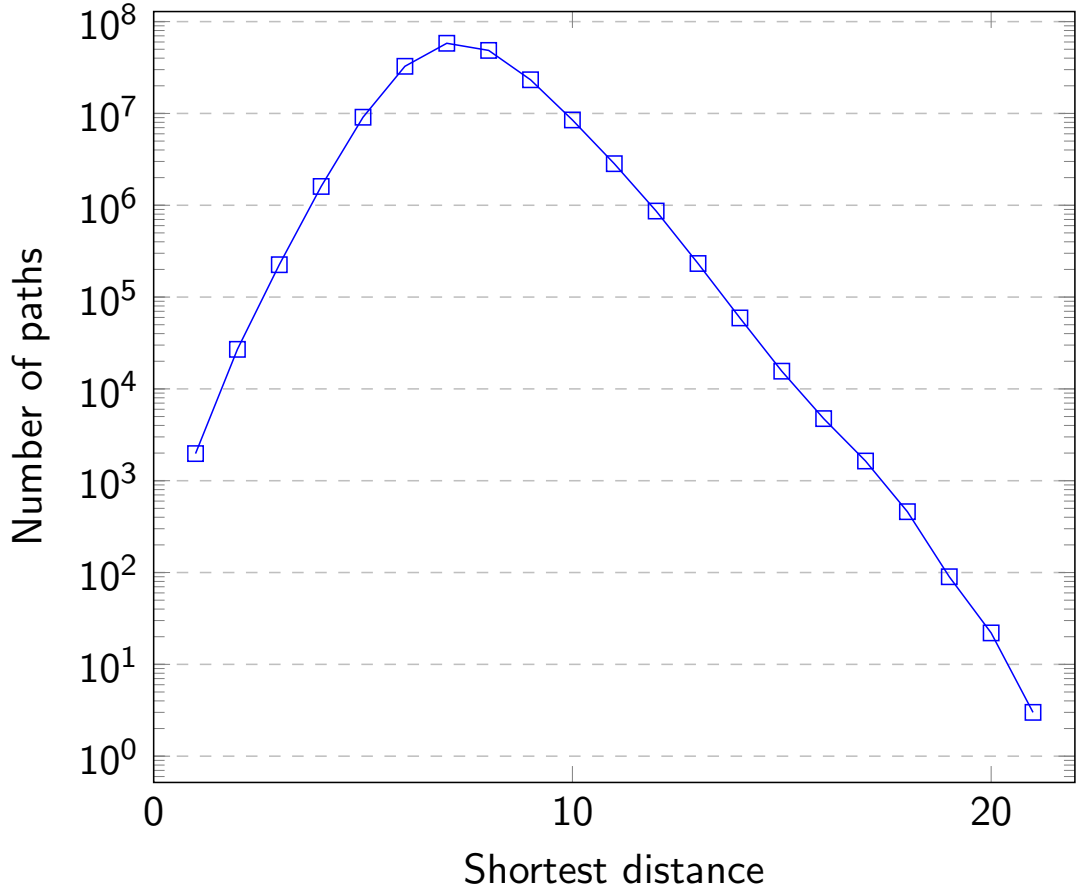


FIGURE – Smallest path distribution for the similarity graph

- Diameter of 21 for an average path of 7.5

Annexes

Similarities

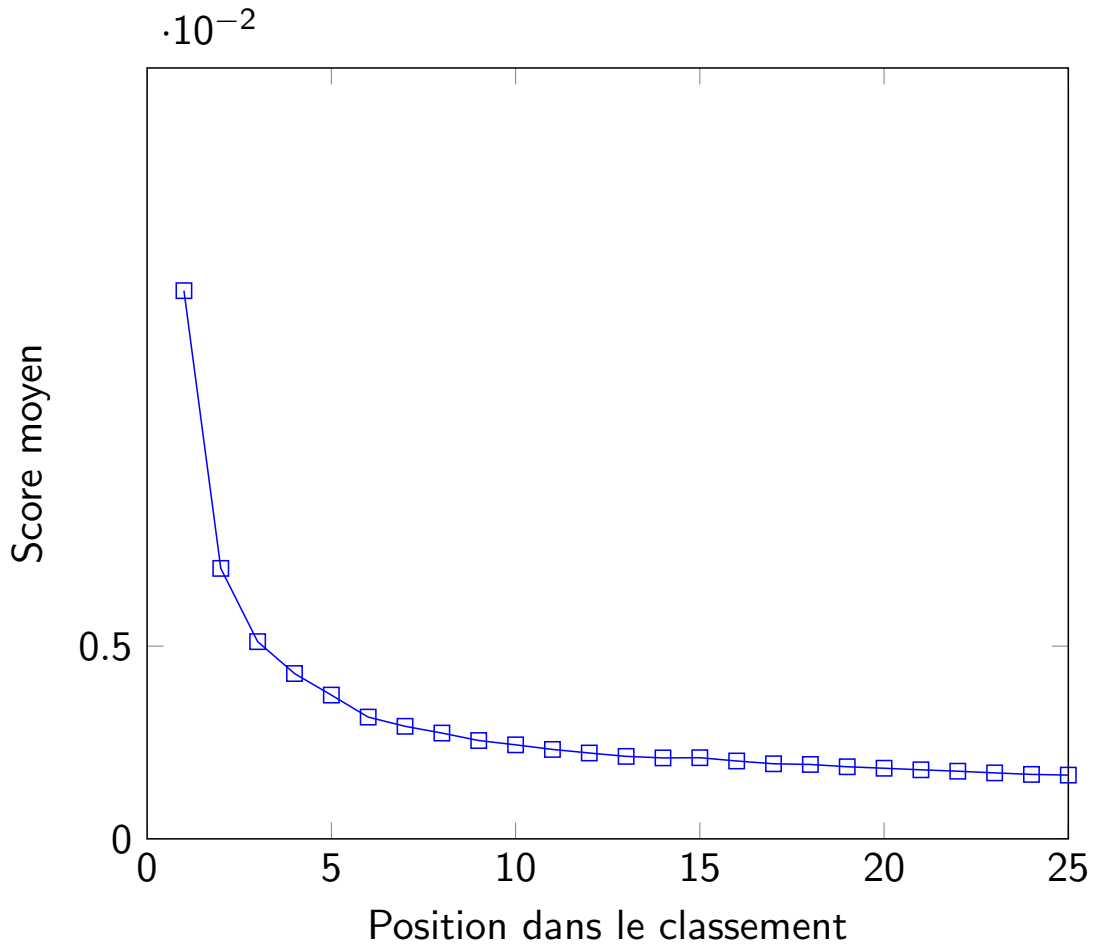


FIGURE – Score similarity evolution

- Really weak scores
- Breaks after the fifth most similar user

Annexes

Intersections

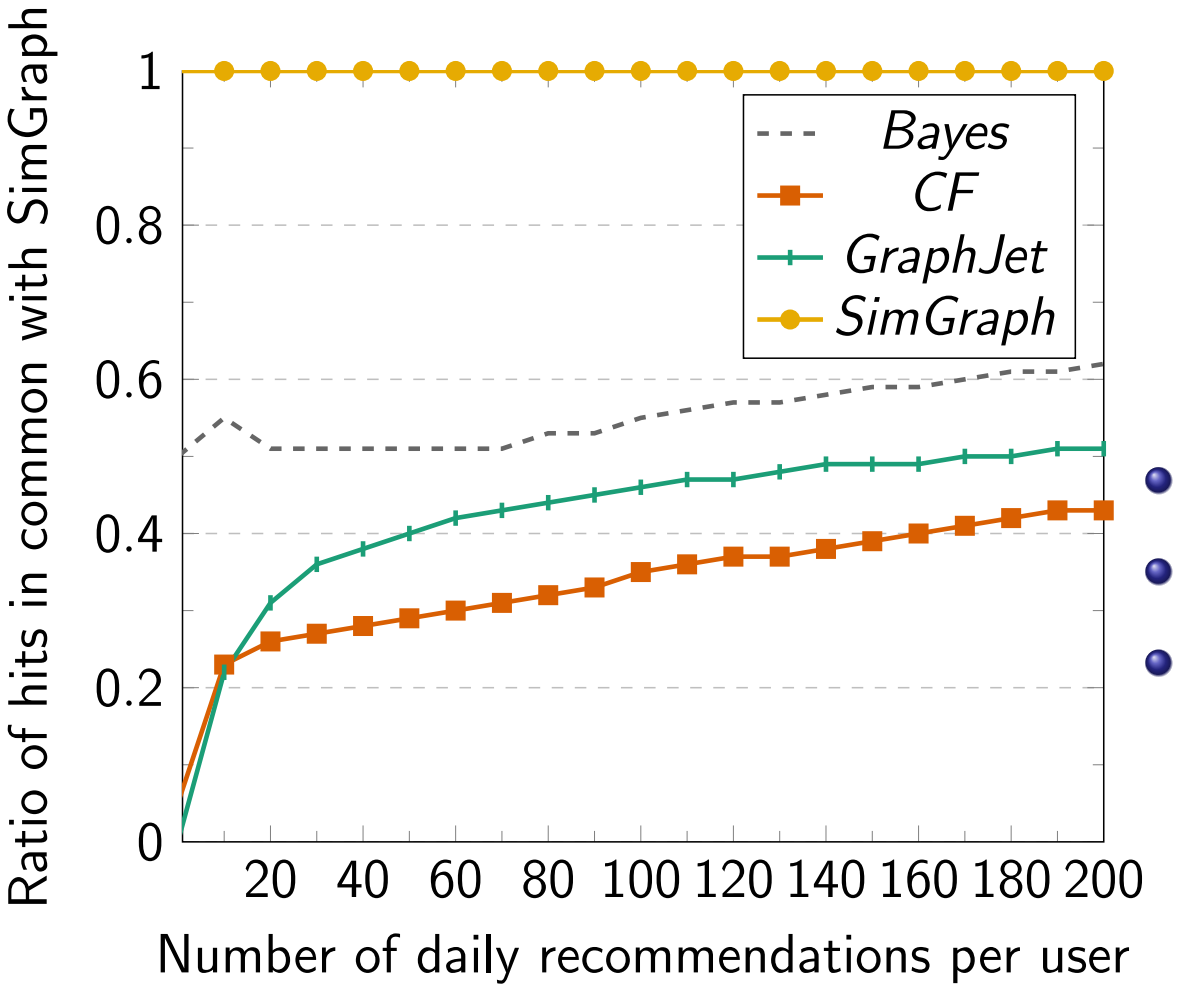
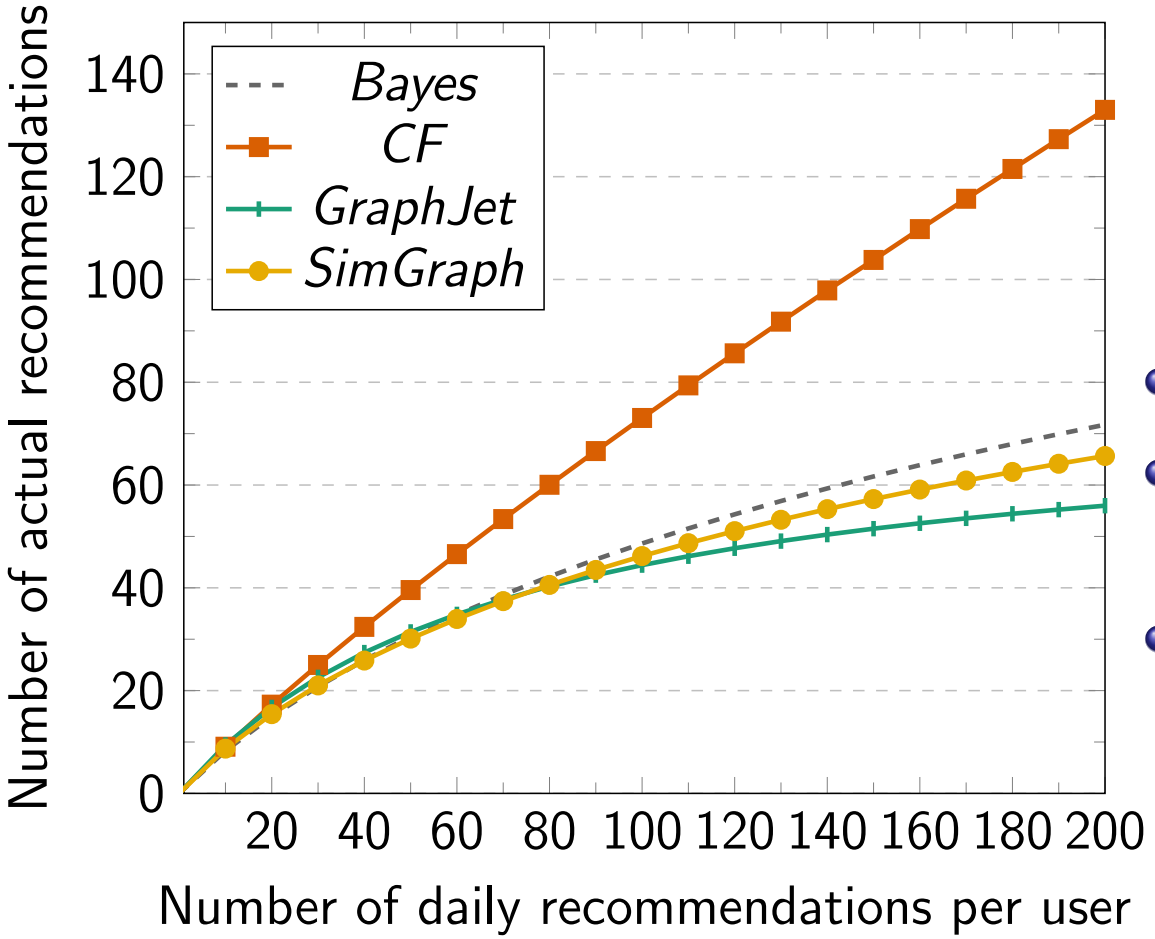


FIGURE – Parts of hits included in SimGraph

Annexes

Number of recommendations



- *CF* is less limited
- Other methods are bunched together
- Threshold effect for *SimGraph* and *Bayes*

FIGURE – Recall capacity