

# **IMPLEMENTACIÓ D'ALGORISMES PER A LA PROPAGACIÓ DE LA INFORMACIÓ EN UNA XARXA SOCIAL EN LÍNIA**

**Puertas Ametller, Pau**

**Curs 2021-22**



**Directors: VLADIMIR ESTIVILL I  
DAVID NETTLETON**

**GRAU EN ENGINYERIA MATEMÀTICA  
EN CIÈNCIA DE DADES**



Universitat  
Pompeu Fabra  
Barcelona

Escola  
d'Enginyeria

**Treball de Fi de Grau**

# IMPLEMENTACIÓ D'ALGORISMES PER A LA PROPAGACIÓ DE LA INFORMACIÓ EN UNA XARXA SOCIAL EN LÍNIA

TREBALL FI DE GRAU DE  
Pau Puertas Ametller

Directors: Vladimir Estivill i David Nettleton

Grau en Enginyeria en Enginyeria Matemàtica  
en Ciència de Dades

Curs 2021-2022



Universitat  
Pompeu Fabra  
*Barcelona*

Escola  
d'Enginyeria

## **Agraïments**

Agrair als tutors per acompanyar-me en aquest camí i ajudar-me a aconseguir el meu objectiu. A la família per donar-me suport en tot moment. Als companys de feina per entendre la situació i adaptar-se a les meves necessitats.

## **Resum**

L'ús de les xarxes socials en línia ha anat augmentat i evolucionant constantment durant la última dècada, donant-nos la possibilitat d'interactuar amb altres persones d'una manera ràpida i senzilla.

L'objectiu d'aquest treball és implementar algorismes que permetin la correcta propagació de la informació en una xarxa social de contacte segons les característiques i interessos de cada usuari. El treball es concentra principalment en com decidir si un usuari concret interacciona amb una publicació determinada en un instant de temps. La part de *Medici* que jo he programat, corresponent a la propagació de la informació, està feta amb Java i JavaFX, i conté un panell interactiu realitzat amb Power Bi. Mitjançant varis algorismes crea les publicacions, propaga la informació, decideix quan els usuaris interaccionen amb aquestes i ens permet analitzar els resultats per extreure conclusions.

## **Resumen**

El uso de las redes sociales en línea ha ido aumentando y evolucionando constantemente durante la última década, dándonos la posibilidad de interactuar con otras personas de una forma rápida y sencilla.

El objetivo de este trabajo es implementar algoritmos que permitan la correcta propagación de la información en una red social de contacto según las características e intereses de cada usuario. El trabajo se concentra principalmente en cómo decidir si un usuario concreto interacciona con una publicación determinada en un instante de tiempo. La parte de *Medici* que yo he programado, correspondiente a la propagación de la información, está hecha con Java y JavaFX y cuenta con un panel interactivo realizado con Power Bi. Mediante varios algoritmos crea las publicaciones, propaga la información, decide cuándo los usuarios interaccionan con éstas y nos permite analizar los resultados para extraer conclusiones.

## **Abstract**

The use of online social networks has been increasing and constantly evolving over the last decade, giving us the possibility to interact with other people quickly and easily.

The objective of this project is to implement algorithms that allow the correct propagation of information in a social contact network according to the characteristics and interests of each user. The project is mainly focused on how to decide if a specific user interacts with a certain post at a given moment of time. The part of *Medici* that I have programmed, corresponding to the propagation of information, is made with Java and JavaFX and has an interactive dashboard made with Power Bi. Through various algorithms, it creates the publications, propagates the information, decides when users interact with them and allows us to analyze the results to draw conclusions.

## Taula de continguts

1.	INTRODUCCIÓ .....	1
1.1	Motivació .....	1
1.2	Objectius .....	1
1.3	Planificació .....	2
2.	ESTAT DE L'ART .....	4
2.1	Context actual de les xarxes socials.....	4
2.2	Estat inicial del treball .....	6
3.	DISSENY DE LA SOLUCIÓ .....	9
3.1	Publicacions .....	9
3.2	Solució estàtica .....	10
3.2.1	Definició .....	10
3.2.2	Creació de publicacions.....	10
3.2.3	Propagació de la informació .....	11
3.3	Solució dinàmica.....	12
3.3.1	Definició .....	12
3.3.2	Creació de publicacions.....	13
3.3.3	Propagació de la informació .....	14
4.	DESENVOLUPAMENT I IMPLEMENTACIÓ .....	17
4.1.	Solució estàtica .....	17
4.2.	Solució dinàmica.....	18
4.3.	Pantalles de la interfície d'usuari.....	21
5.	VALIDACIÓ DE LA SOLUCIÓ.....	23
5.1.	Estadístiques de xarxes socials reals.....	23
5.2.	Visualització de dades amb Power Bi.....	24
5.3.	Estadístiques Medici .....	25
6.	CONCLUSIONS .....	30
7.	BIBLIOGRAFIA.....	31
8.	ANNEXOS .....	33
	Annex A: Propagació de la informació en el disseny de la solució dinàmica .....	33
	Annex B: Pseudo-codi de la solució estàtica .....	37
	Annex C: Càcul de les distàncies 2 i 3 en la versió dinàmica .....	39
	Annex D: Pseudo-codi de la solució dinàmica .....	44
	Annex E: Propagació de la informació amb la versió dinàmica .....	49

Annex F: Taulell Power BI.....	59
Annex G: Fitxers d'entrada del programa .....	63
Annex H: Fitxers de sortida del programa.....	65
Annex I: Explicació de les pantalles de la interfície d'usuari.....	69
Annex F: Més execucions del programa.....	75

## Taula d'ilustracions

1.3.1. Diagrama de Gantt.....	3
2.2.1. Seqüència de processament .....	7
2.2.2. Generació de dades .....	8
3.1.1. Exemple publicació .....	9
3.2.1. Diagrama solució estàtica.....	10
3.2.2. Funció long tail.....	11
3.2.3. Pseudocodi Solució Estàtica.....	12
3.2.4. Propagació informació solució estàtica .....	12
3.3.1. Diagrama solució dinàmica .....	13
3.3.2. Pesudocodi Solució Dinàmica .....	15
3.3.3. Propagació informació solució dinàmica .....	16
4.3.1. Interficie de l'usuari - Generate Data.....	21
4.3.2. Interficie de l'usuari - Results Publications .....	22
5.1.1. Publicacions i Engagement Rate Facebook i Instagram.....	23
5.1.2. Engagement Rate per seguidors Instagram i Twitter.....	24
5.2.1. Publications Overview Power BI .....	25
5.3.1. Execució amb els paràmetres per defecte .....	26
5.3.2. Interaccions a l'instant 145 .....	26
5.3.3. Topic, type and community of publication 1916.....	27
5.3.4. Característiques dels dos usuaris .....	27
5.3.5. Interaccions a distàncies 2 i 3 .....	27
5.3.6. Execució amb una xarxa més gran .....	28
A. Nodes i enllaços.....	33
A. Nova publicació .....	33
A. Rebuda de publicacions a distància 1 .....	34
A. Interaccions a distància 1 .....	34
A. Rebuda de publicacions a distància 2 .....	35
A. Interaccions a distància 2 .....	35
A. Rebuda de publicacions a distància 3 .....	36
A. Interaccions a distància 3 .....	36
A. Graf final.....	36
D. Graf de l'exemple .....	49
D. Creació publicació 1 .....	52
D. Propagació distància 1 .....	52
D. Decisió interacció .....	55
D. Propagació informació usuari O .....	55
D. Decisió d'interacció.....	56
D. Propagació informació usuaris E i H .....	57
D. Decisió interacció .....	57
D. Resultat final.....	58
E. Power BI - Stats Overviwe.....	59
E. Power BI - User Overview .....	60
E. Power BI - User Overview amb filtre .....	60
E. Power BI - Publications Overview .....	61

E. Power BI - Engagement Overview .....	61
E. Power BI - Publications comparition .....	62
F. Topics description .....	63
F. Graph Structure .....	64
F. Community Labels .....	64
G. kby30k_out .....	65
G. 1kby30_out1 .....	65
G. 1kby30_out2 .....	65
G. 1kby30_outG .....	66
G. Info_publications_din .....	66
G. Numfriends_numpub_din .....	66
G. Score_given_publications .....	67
G. Tagged_publications_din .....	67
G. User_publications_din .....	67
G. User_interactions_publications_din .....	68
H. Interficie d'usuari – Portada .....	69
H. Interficie d'usuari - Generate Graph .....	70
H. Interficie d'usuari - Input file Settings .....	70
H. Interficie d'usuari - User Attributes .....	71
H. Interficie d'usuari - Profile Seed Settings .....	71
H. Interficie d'usuari - Communities & Profiles .....	72
H. Interficie d'usuari - Output file settings .....	72
H. Interficie d'usuari - Advanced Settings .....	73
H. Interficie d'usuari - Generate Data .....	73
H. Interficie d'usuari – Results .....	74
H. Interficie d'usuari - Results Publications .....	74
I. Execució 1 .....	75
I. Execució 2 .....	76
I. Execució 3 .....	77
I. Execució 4 .....	78
I. Execució 5 .....	79
I. Execució 6 .....	80
I. Execució 7 .....	81

# **1. INTRODUCCIÓ**

## **1.1 Motivació**

L'ús de les xarxes socials està molt estès entre la societat. Un 58,5% de la població mundial utilitza aquest tipus de xarxes i la mitjana de temps diari invertit és de 2 hores i 27 minuts per usuari [1].

Des de ben petit m'han cridat molt l'atenció les xarxes socials i sempre he estat un usuari actiu i curiós d'aquestes. Per sort o per desgràcia, tenen una importància vital en la societat, i cada cop un grup més ampli en fa un ús extens.

La possibilitat de realitzar simulacions de xarxes socials em va captivar molt l'atenció i l'opció d'intentar entendré en què es basen els usuaris a l'hora d'interactuar amb les diferents publicacions va ser un tema que des del primer moment em va resultar molt interessant. Cada xarxa social és un món completament diferent que et permet realitzar diverses accions i adquirir múltiples experiències. Però si alguna cosa tenen en comú totes les xarxes socials és que en totes es crea informació i es produeix la propagació entre els diferents usuaris que la formen.

Pel que fa a la basant matemàtica, les xarxes socials oculten un rerefons d'algorismes, prediccions i analítica molt captivador. Absolutament tota la informació que es mostra està ordenada i classificada i cap detall està vinculat amb l'atzar.

És per això que vaig trobar la implementació d'algorismes per a la propagació d'informació en una xarxa social en línia un tema immensament interessant que em permetria adquirir grans coneixements i aprofundir en un aspecte fonamental del meu dia a dia.

## **1.2 Objectius**

### **a) Objectius del projecte**

Per una banda, entendre el funcionament de la propagació d'informació en les xarxes socials reals i en què es basa la gent per interaccionar o no amb les publicacions.

D'altra banda, implementar algorismes que permetin la propagació de la informació en una simulació d'una xarxa social. Aquesta propagació ha de ser el més proper a la realitat possible. Per a dur-ho a terme es realitza la simulació mitjançant un programa desenvolupat amb Java i JavaFX que permet a l'usuari interaccionar i modificar una àmplia varietat de paràmetres. La idea és que el programa pugui ser utilitzat per tercers de mode experimental per a extreure les seves pròpies conclusions.

Per últim, poder analitzar i comprovar que la simulació s'ha realitzat correctament, visualitzant la informació que s'ha generat usant tècniques de visualitzacions de dades.

El projecte fet en Java/JavaFX i el programa fet amb Power BI es poden trobar al següent enllaç: <https://github.com/dnettlet/MEDICIP>

### b) Objectius personals

Primerament, com a usuari habitual de les xarxes socials, entendre correctament els diferents passos que es produeixen en la propagació de la informació.

Com a fan de la programació i persona inquieta, programar un programa d'aquesta envergadura en un llenguatge que no domino com és Java i JavaFX.

Darrerament, aprofundir en el tema de les visualitzacions de dades i aplicar-ho a grafs.

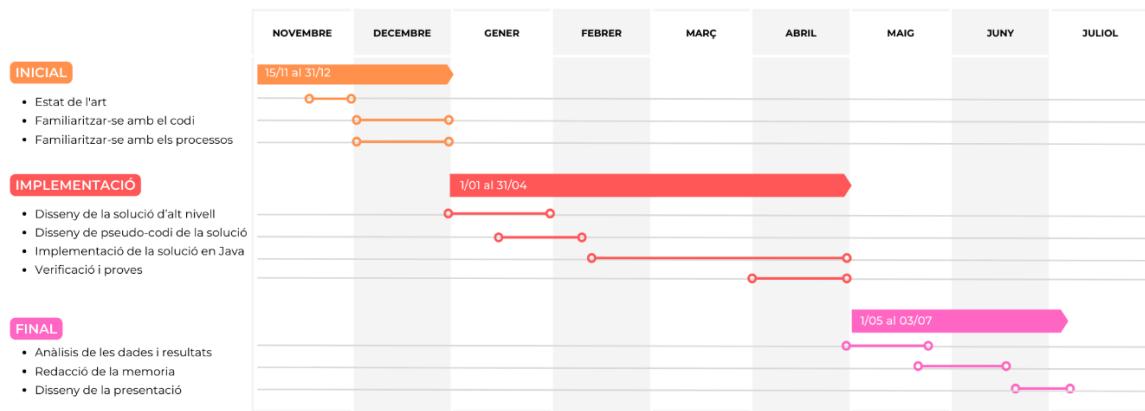
## 1.3 Planificació

La planificació del projecte és una part estrictament important. Dividir-lo en varíes fases i establir intervals temporals permetrà realitzar les tasques en el temps correcte. S'ha de tenir en compte que la planificació s'ha de seguir de manera estricta, però al mateix temps permetre's una flexibilitat per a possibles problemes que puguin sorgir durant la implementació d'aquest.

El digrama que s'ha seguit ha estat el següent:

	Descripció de la tasca	Temps d'entrega	Fase
1	Estat de l'art: Com està el treball actualment i context actual de les xarxes socials.	15 nov – 30 nov	Fase inicial
2	Familiaritzar-se amb el codi	1 des – 31 des	
3	Familiaritzar-se amb els processos i el flux de treball de l'aplicació	1 des – 31 des	
4	Disseny de la solució d'alt nivell: decidir que és el que es voldrà implementar .	1–31 gen	Fase d'execució
5	Disseny de pseudo-codi de la solució	15 gen – 15 feb	
6	Implementació de la solució en Java i JavaFX	16 feb – 31 abr	
7	Verificació i proves	1 -31 abr	
8	Anàlisis de les dades i estudi dels resultats	1 mai – 19 mai	Fase final
9	Redacció de la memòria	20 mai – 16 jun	
10	Disseny de la presentació	17 jun – 4 jul	

Amb la corresponent visualització al llarg del temps:



1.3.1. Diagrama de Gantt

## **2. ESTAT DE L'ART**

### **2.1 Context actual de les xarxes socials**

L'ús de les xarxes socials en línia ha anat augmentat i evolucionant constantment durant l'última dècada. Amb les xarxes socials tenim la possibilitat d'interactuar amb altres persones i es van constraint amb el que cada persona aporta a la xarxa. Cada nou usuari que ingressa transforma el grup en un de nou i la xarxa no és la mateixa si un membre deixa de ser part.

Algunes dades recollides el gener de 2022 [2] que ens poden fer entendre la dimensió i la importància en el dia a dia de les persones són les següents:

- Hi ha un total de 4,62 bilions d'usuaris en les xarxes socials. El que es tradueix en un 58,4% de la població. Aquest percentatge és molt major quan ens fixem únicament en Europa o Nord Amèrica.
- En els darrers 12 mesos, 424 milions de nous usuaris s'han registrat a alguna xarxa social.
- Els usuaris dediquen de mitjana 2 hores i 27 minuts diaris a les xarxes.
- El nombre mig de xarxes socials que utilitza una persona durant un mes és de 7,5.
- Les plataformes més usades al món són Facebook amb 2.910 milions d'usuaris, seguida de Youtube, Whatsapp i Instagram amb 2.562, 2.000 i 1.478 respectivament.
- En un estudi fet al 2021 a Anglaterra sobre l'ús de les xarxes socials de nens i pares [3] es va extreure que el 44% dels nens entre 8 i 11 anys utilitza xarxes socials. La dada incrementa fins al 87% en nens d'entre 12 i 15 anys.

Existeixen diversos tipus de xarxes socials [4, 5] segons la utilitat i l'objectiu dels usuaris en aquestes. Els principals són:

- Xarxes socials de contacte: Permeten connectar persones amb interessos i antecedents similars i potencien la comunicació i el contacte entre els usuaris. Els continguts que es comparteixen són fotos i vídeos. Es creen enllaços entre els usuaris, que poden ser dirigits o no dirigits. Els continguts es compartiran amb els usuaris enllaçats. Exemples són Facebook, Instagram o LinkedIn.
- Xarxes per compartir continguts: Usades per buscar i compartir fotografies, vídeos, vídeos en directe i altres tipus de continguts. Permeten als usuaris descobrir i compartir arxius a una gran audiència. No necessàriament ha d'existir un enllaç entre usuaris per poder veure els continguts. Exemples són Youtube, Twitch o Twitter. Aquest darrer es podria considerar una barreja amb les xarxes socials de contacte.
- Fòrums de discussió: Aquest tipus de xarxes socials són emprades per buscar, compartir i discutir diferent tipus d'informació, opinions i notícies. Qualsevol usuari

pot participar i interactuar en els fòrums que es creen. Exemples són Stack Overflow, Reddit o Quora.

- Bloggs i xarxes de publicacions: Serveixen per publicar, descobrir i comentar articles i bloggs. Generalment, són de caràcter personal, amb una estructura cronològica que es va actualitzant regularment i es dedica a un tema en concret. Qualsevol usuari pot accedir als articles i comentar les seves opinions. Exemples són WordPress o Tumblr. Dins d'aquestes, trobem el subtipus de xarxes de publicacions acadèmiques, com ResearchGate i Academia.
- Xarxes de reviews dels consumidors: Els usuaris poden valorar, compartir i comentar sobre una gran varietat de productes, serveis o marques. Un clar exemple és TripAdvisor, on els usuaris poden valorar restaurants d'arreu del món segons el servei obtingut. Qualsevol usuari pot donar la seva opinió i són molt útils per fer-se una idea del producte o servei que consumiràs.
- Hi ha altres tipus de xarxes socials, com ara xarxes socials de compres, de marcadors i conservació de continguts, xarxes basades en interessos o xarxes socials per lligar. Un altre tipus són les xarxes per a compartir codi, amb GitHub com a principal exemple.

En el cas d'aquest treball es treballarà amb les xarxes socials de contacte amb enllaços no dirigits.

A part del creixement de les xarxes socials, també ha augmentat el desig d'entendre i analitzar que és el que succeeix en aquestes. Per tal d'extraure el màxim d'informació de les dades que es generen a les xarxes socials ha aparegut el que es coneix com a *Social Data Analysis* [6]. En grans trets és el procés d'investigació d'estructures socials mitjançant l'ús de xarxes i teoria de grafs. Les estructures en xarxa es caracteritzen per nodes, que podrien ser actors individuals o persones, i enllaços que els connecten, que serien les relacions o interaccions entre els nodes. Aquest procés ens permet processar una gran quantitat de dades relacionals i descriure l'estructura general de la xarxa relacional. Mitjançant la selecció de termes i paràmetres es poden confirmar els nodes influents a la xarxa. I gràcies a l'anàlisis de nodes, clústers i relacions es pot descriure clarament l'estructura de comunicació i quina és la posició dels diferents individus.

És per això, que han sorgit noves regles de comportament sobre com els usuaris tendeixen a agrupar-se per afinitats i interessos, com estan interconnectats entre ells, com capturar i avaluar l'activitat, quines són les dades demogràfiques clau, com es produeix la propagació de la informació i la dinàmica general del que fa a les xarxes socials funcionar correctament.

## 2.2 Estat inicial del treball

L'aplicació *Medici* [7,8,9], desenvolupada per en David F. Nettleton, en Sergio Nettleton i en Marc Canal i Farriol, es tracta generador de dades sintètic per a grafs de xarxes socials en línia. Aquesta permet acumular dades d'usuaris i pot ser de gran utilitat per a la investigació aplicada i estudis de població en altres camps. És utilitzat en la mineria de dades [10].

*Medici* és una aplicació accessible per a usuaris de nivell mitja en programació, tot i que també pot ser utilitzada com una eina de recerca per a usuaris d'alt nivell. Conté una interfície visual molt intuïtiva que permet rebaixar l'entrada a usuaris de baix nivell, facilitant que pràcticament qualsevol usuari pugui utilitzar el programa i aconseguir les dades que necessita.

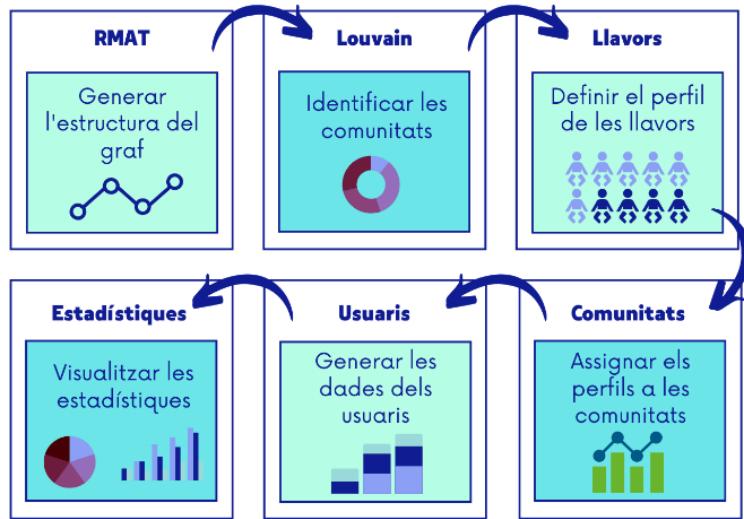
L'aplicació consta d'un algorisme principal que s'encarrega d'assignar i propagar les dades per l'estructura del graf. Aquest utilitza dos algoritmes de tercera, anomenats RMAT i Louvain.

RMAT és un model recursiu de mineria de grafs. Aquest té l'objectiu de modelar i generar l'estructura d'un graf, obtenint així la seva parametrització en termes de variables descriptores donades.

Louvain és un mètode que té l'objectiu d'etiquetar les comunitats. Primerament, fa una cerca per petites comunitats optimitzant la modularitat de forma local. A continuació, agrega nodes de la mateixa comunitat i construeix una nova xarxa els nodes de la qual són les comunitats. Aquests dos passos es repeteixen iterativament fins que el valor de modularitat és maximitzat. L'actual versió de *Medici* ha retocat aquest mètode per tal de produir exactament 10 comunitats.

L'algorisme Medici per a la generació del graf segueix una estructura seqüencial molt intuïtiva.

A continuació podem veure el workflow que segueix l'algorisme.



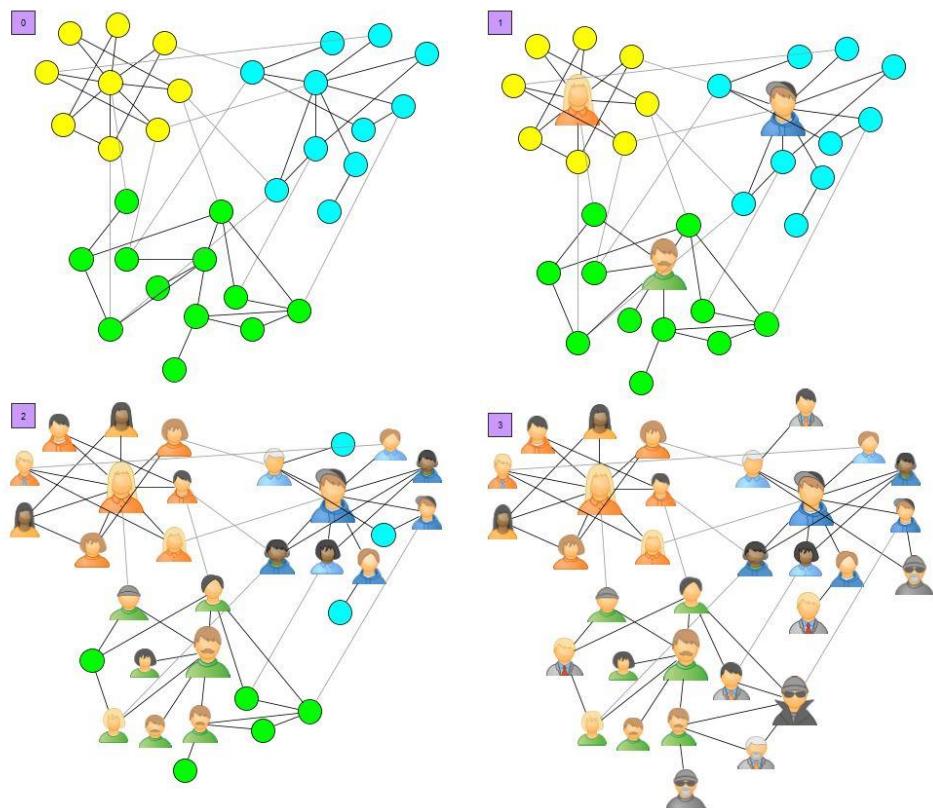
#### 2.2.1. Seqüència de processament

Primerament s'utilitzen els algorismes descrits anteriorment per tal de generar l'estructura del graf i per identificar les comunitats. A continuació es defineixen els perfils de les llavors i s'assignen aquests perfils a les diferents comunitats. Arribats a aquest punt, es generen les dades dels usuaris. Finalment, es poden comprovar les estadístiques generades a partir del graf i usuaris generats.

La generació de dades es fa mitjançant els següents passos:

1. El primer pas és triar quins nodes seran les llavors de cada comunitat. Cada node llavor ha d'estar com a mínim a distància 3 de qualsevol altre node llavor. A més, s'ha d'assignar el prototipus de perfil per a les llavors de cada comunitat.
2. El següent pas és assignar dades als veïns de cada llavor en funció del perfil de la llavor, ja que normalment, els veïns són similars a les llavors. Per tant, les dades tindran un component de similitud, amb els veïns que tenen dades assignades, i un component aleatori, per promoure un grau de diversitat entre els usuaris.
3. L'últim pas necessari serà assignar dades als nodes restants, és a dir, aquells que no han estat processats als passos anteriors. Es farà de la mateixa manera que s'ha descrit al pas anterior, amb un component de similitud i un component aleatori.

A continuació veiem els 3 passos en un exemple amb 3 comunitats:



#### 2.2.2. Generació de dades

És doncs amb tot el procediment descrit com s'aconsegueix generar el graf i les dades dels usuaris per a simular una xarxa social. Veiem com tots els usuaris de l'exemple a petita escala tenen assignats els seus perfils. Cal destacar que en aquest exemple simplificat només hi havia una llavor per comunitat, però quan es desenvolupa el programa, les comunitats tenen vàries llavors per a propagar més de pressa la informació.

### 3. DISSENY DE LA SOLUCIÓ

#### 3.1 Publicacions

Per a poder portar a terme la propagació de la informació en la simulació de la xarxa social, serà necessària la generació prèvia d'aquesta informació. En el cas de *Medici* es farà mitjançant publicacions que crearan els usuaris de la xarxa. Cada publicació tindrà un identificador únic i podrà ser de tres tipus: imatges, vídeos i texts.

A part, cada publicació tindrà una temàtica especificant el tema de cada publicació. Les temàtiques de les quals poden ser les publicacions són: artistes de música, marques de begudes, televisió, esports, animals, videojocs, notícies i present, menjar i viatges. Cada publicació tindrà a més una breu descripció del que tracta, que anirà directament relacionada amb la temàtica. Per exemple una publicació de temàtica animal pot tenir la següent descripció: *gat dormint al llit*.

No totes les publicacions són igual de bones, i és per això que cada publicació tindrà associat un valor numèric per definir la qualitat d'aquesta. També, com es produeix a la majoria de xarxes socials, diversos usuaris podran ser etiquetats a cadascuna de les publicacions. Un exemple de publicació seria el següent:



3.1.1. Exemple publicació

Un cop es vagi creant la informació, aquesta s'anirà propagant per la xarxa. Els usuaris que vegin cadascuna de les publicacions tindran la possibilitat d'interactuar amb aquestes. En la simulació *Medici* hi ha tres tipus d'interaccions possibles:

- Fer *like*: per indicar que li agrada la publicació.
- Deixar un comentari: per fer pública la seva opinió envers la publicació.

- Compartir: per tal de fer arribar la publicació als usuaris que estiguin connectats amb ell.

*Medici* permetrà a l'usuari fer la propagació de la informació en dos modes: mode estàtic i mode dinàmic.

## 3.2 Solució estàtica

### 3.2.1 Definició

El primer mode és simple, però al mateix temps permet a l'usuari entendre com es produeix la propagació de la informació en un model senzill i de baix nivell. A part, per tal de poder fer el correcte desenvolupament de la solució dinàmica, la solució estàtica permet rebaixar la dificultat i actuar com un pas entremig cap a la solució final.

En aquesta solució les interaccions es produiran de forma estàtica, és a dir, en el mateix interval temporal i no utilitzaran la informació de les interaccions que s'hagin produït anteriorment. A l'inici del programa es crearan totes les publicacions i es produiran les interaccions entre aquestes i els usuaris. L'única informació que es tindrà en compte a l'hora de decidir si un usuari concret interactua amb una publicació és la informació de l'usuari (tots els atributs que es generen d'aquest quan es crea el graf) i la informació de la publicació.

En aquest mode les úniques interaccions possibles per part dels usuaris són donar *like* i fer comentaris, i no es permet que es comarteixin publicacions. De la mateixa manera tampoc es permet que s'etiqueten amics en aquestes.



3.2.1. Diagrama solució estàtica

### 3.2.2 Creació de publicacions

En el moment d'executar el programa, cada usuari publicarà zero, una o més d'una publicació. A les xarxes socials ens trobem en el cas que pocs usuaris publiquen molt i

molts usuaris publiquen poc. Per a representar aquest fet el programa ho farà mitjançant una probabilitat que vindrà donada per una distribució *long tail* [11, 12]. Aquest tipus de distribucions poden ser expressades per un exponent  $\alpha$ , i una constant C:

$$p(x) = Cx^{-\alpha}$$

La funció  $p(x)$  és expressada sobre un interval  $[min, \infty)$  i l'exponent  $\alpha$  determina la llargada de la cua. Aquestes distribucions, a diferència de les distribucions normals, no es centren al voltant d'un punt mig. Són totalment asimètriques.

En cas de la solució estàtica s'ha adaptat una funció *long tail* per generar probabilitats properes a la realitat de les xarxes socials. Utilitzant  $C=0.8$ ,  $\alpha=1$  i  $min=1$  obtenim el següent:

$$p(x) = 0.8x^{-1}$$



### 3.2.2. Funció long tail

Els eixos en el nostre cas equivaldrien a:

- Eix de les x: nombre de publicacions que fa un usuari
- Eix de les y: probabilitat que té de fer-les.

Per tant, mitjançant aquesta funció aconseguiríem les següents probabilitats de fer cap, una o més publicacions per a cadascun dels usuaris:

- 80% de fer la primera publicació.
- 40% de fer la segona publicació (en el cas que hagi fet prèviament la primera).
- 26% de fer la tercera publicació (en el cas que hagi fet prèviament la segona).
- 20% de fer la quarta publicació (en el cas que hagi fet prèviament la tercera).
- 16% de fer la cinquena publicació (en el cas que hagi fet prèviament la quarta).
- 13% de fer la sisena publicació (en el cas que hagi fet prèviament la cinquena).
- 11,4% de fer la setena publicació (en el cas que hagi fet prèviament la sisena).
- ...

### 3.2.3 Propagació de la informació

Un cop s'han generat les publicacions, els usuaris tindran l'oportunitat d'interaccionar amb les publicacions dels seus amics. L'algorisme complet utilitzat per decidir si fa like, like+comentari o res, serà explicat a l'apartat: *4.1. Solució estàtica*.

De forma general és el següent:

---

**Algorithm 1:** Static Solution

---

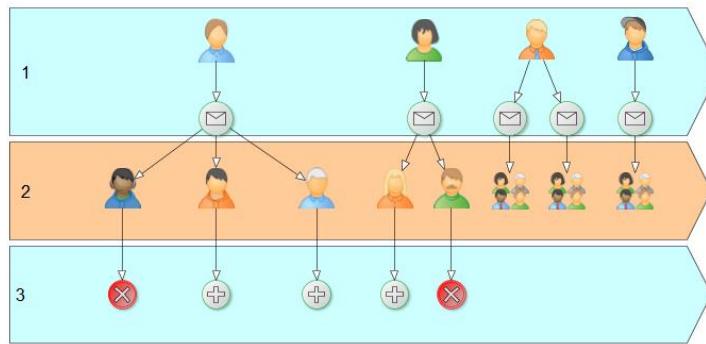
```
create graph;
for user in users do
    | create publications;
end
for publication in publications do
    for friend in user friends do
        | calculate score;
        | decide interaction with this score;
    end
end
create output files;
```

---

### 3.2.3. Pseudocodi Solució Estàtica

Per entendre-ho, veiem un exemple. En aquest cas, la informació es propaga de la següent manera:

1. Cada usuari crea entre 0 i x publicacions
2. Cada publicació arriba als usuaris que són amics del creador, és a dir, els que estan connectats amb ell en el graf.
3. Mitjançant un algorisme es defineix una puntuació, anomenada *score*, entre 0 i 1 que permet decidir si l'usuari fa like, like i comentari o deixa passar la publicació.
4. Un cop fetes totes les interaccions acaba el programa i es generen els fitxers de sortida.



### 3.2.4. Propagació informació solució estàtica

## 3.3 Solució dinàmica

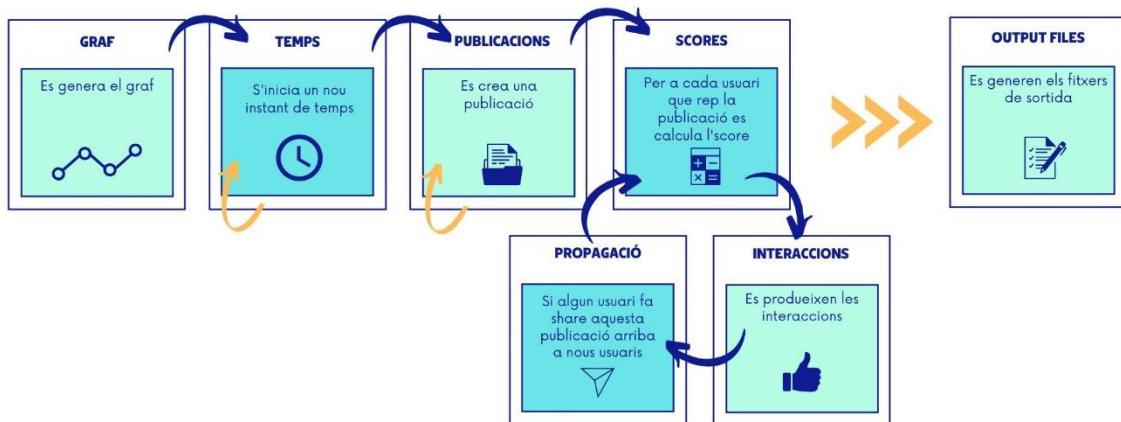
### 3.3.1 Definició

En aquest mode es desenvoluparà una solució que permetrà crear publicacions i fer interaccions en temps real durant un interval de temps. Això permetrà, entre altres coses, que les accions passades dels usuaris es tinguin en compte.

Els interessos inicials de l'usuari és un punt molt important a considerar, ja que inicialment les publicacions amb les quals interacció s'adaptaran als gustos i aficions de l'usuari. Tot i això, els interessos que ha definit l'usuari en el moment de crear el seu compte poden estar incomplets, ser incorrectes o simplement antiquats. Per tant, també serà molt rellevant esbrinar els interessos i aficions reals (i actuals) de l'usuari mitjançant les interaccions que hagi realitzat amb publicacions anteriors.

Per exemple, si l'usuari A li ha donat *like* a 3 publicacions de l'usuari B, i l'usuari B penja una quarta publicació, aleshores la probabilitat que l'usuari A interacció amb aquesta publicació serà molt alta. De la mateixa manera si l'usuari A està contínuament interactuant amb les publicacions d'esports i obviant les publicacions de viatges, el més probable és que segueixi amb la mateixa dinàmica envers les publicacions d'aquestes dues temàtiques.

En aquest mode, sí que s'incorporà la possibilitat per part dels usuaris de compartir les publicacions. Quan un usuari comparteix, fa que la publicació arribi a tots els seus amics, independentment de si són amics o no del creador d'aquesta. Això farà que se'ls hi doni pes als anomenats *influencers* (usuaris amb molts amics), que provocaran que les publicacions arribin a un públic molt més ampli.



3.3.1. Diagrama solució dinàmica

### 3.3.2 Creació de publicacions

Les publicacions s'aniran creant a mesura que va avançant el temps del programa. A cada instant es faran entre 0 i K publicacions. Per defecte està configurat per haver-hi un màxim de 25 publicacions en cada instant temporal.

En aquest mode també es permet que una publicació tingui altres usuaris etiquetats. Únicament poden ser etiquetats usuaris que siguin amics del creador. Per defecte poden ser etiquetats un màxim de 10 usuaris.

### 3.3.3 Propagació de la informació

#### a) Conceptes clau

Alguns conceptes clau que ens ajudaran a entendre el que es produeix en aquest mode i com delimitar les interaccions són els següents:

En teoria de grafs la distància entre dos vèrtexs és el nombre d'arestes que hi ha en el camí més curt possible [13].

La teoria dels sis graus de separació [14] estableix que qualsevol habitant de la Terra podria conèixer qualsevol persona del món amb un màxim de sis o menys connexions mútues entre ells i una altra persona. Ja sigui a través de coneguts, amics o membres de la seva família. Aquesta teoria es pot aplicar també en grafs, i ens serà útil per poder limitar les nostres interaccions.

#### b) Solució

La idea d'aquesta solució és seguir un ordre temporal per a les publicacions i interaccions que es vagin produint. A més, l'usuari que interacció amb una publicació no sempre serà amic de l'usuari que la creï, ja que en aquest mode es poden compartir publicacions, i per tant, es produiran interaccions a diferents distàncies del node creador. S'adaptarà una mica el concepte i es parlarà de distància referint-se al camí que ha fet la publicació fins que l'usuari ha interaccionat. És a dir tindrem els següents casos:

- Interacció a distància 1: l'usuari interacciona en el moment en el qual veu la publicació directament pel creador.
- Interacció a distància 2: l'usuari interacciona en el moment en què un altre usuari ha compartit la publicació a distància 1. Es pot produir tant si és amic del creador com si no.
- Interacció a distància 3: l'usuari interacciona en el moment en què un altre usuari ha compartit la publicació a distància 2. Es pot produir tant si és amic del creador com si no.
- ...

Com la simulació es farà per a petites i mitjanes xarxes socials, només es podran fer interaccions de fins a distància 3.

L'algorisme utilitzat per decidir si fa like, like+comentari o res, serà explicat a l'apartat 4.2. *Solució dinàmica*. De forma general, el pseudocodi en molt alt nivell és el següent:

---

**Algorithm 1:** Dynamic Solution

---

```
create graph;
for time in max time do
    for user in max users by time do
        user create a publication;
        for friend1 of user friends do
            calculate score;
            decide interaction for friend1 with this score;
            if friend1 do share then
                for friend2 of friend1 friends do
                    calculate score;
                    decide interaction for friend2 with this score;
                    if friend2 do share then
                        for friend3 of friend2 friends do
                            calculate score;
                            decide interaction for friend3 with this score;
                    end
                end
            end
        end
    end
end
create output files;
```

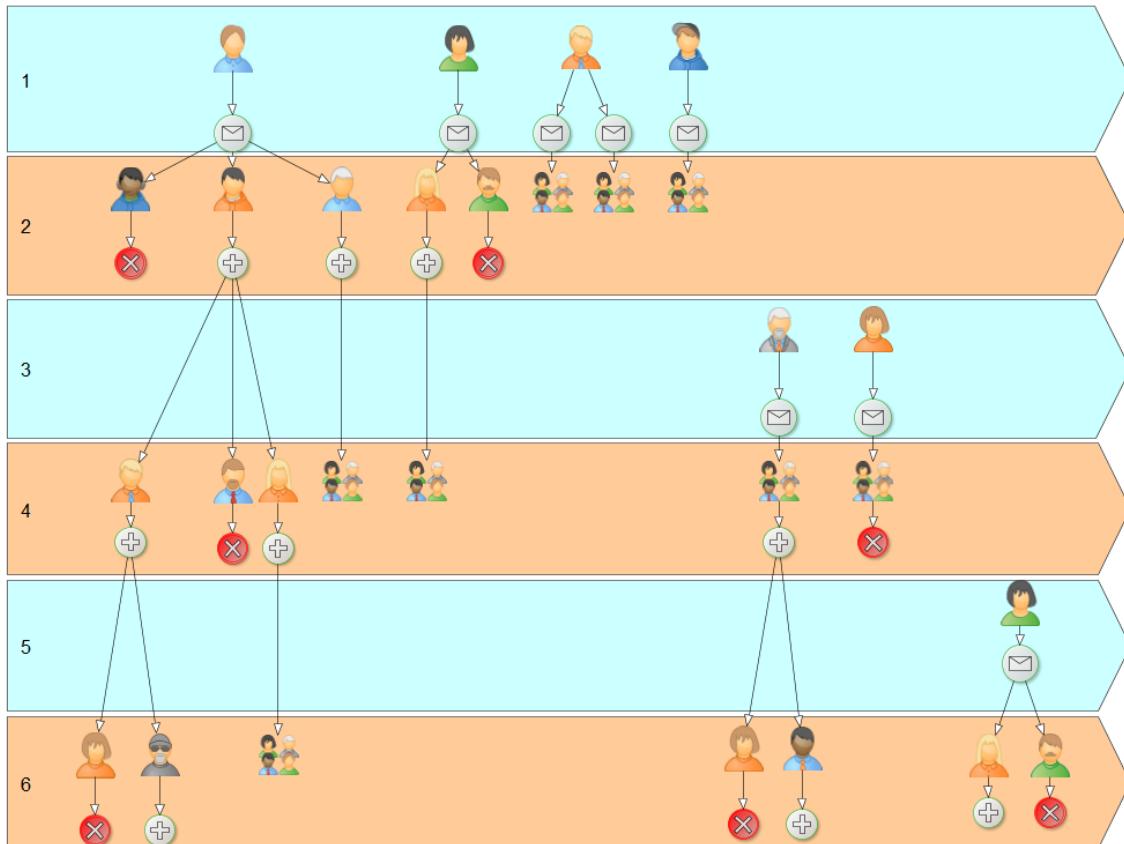
---

### 3.3.2. Pesudocodi Solució Dinàmica

Per entendre-ho en funcionament, a continuació s'especifica un exemple de com es propaga la informació:

1. Un nombre aleatori entre 0 i k1 publicacions seran creades per n1 usuaris en primera instància.
2. A continuació les interaccions de distància 1 es produiran.
3. En aquest instant es crearan k2 noves publicacions per n2 usuaris i es propagarà la informació de distància 1 de les primeres k1 publicacions.
4. Immediatament, es produiran dos esdeveniments: començaran a ser interaccionades amb distància 1 les noves publicacions i amb distància 2 les publicacions antigues.
5. A continuació es creen k3 noves publicacions per n3 usuaris i es propaga la informació anterior.
6. Aleshores es produeixen les interaccions de distància 1 de les noves k3 noves publicacions, les de distància 2 de les k2 publicacions i les de distància 3 de les primeres k1 publicacions.
7. Així es va produint successivament, on les publicacions com a molt seran interaccionades a distància 3.

En la següent il·lustració es mostren els diferents passos. El símbol + indica que hi ha interacció entre l'usuari i la publicació, el símbol X indica que no hi ha cap interacció:



### 3.3.3. Propagació informació solució dinàmica

Per entendre com es produeix la propagació de la informació en un graf com el que tindrem en la simulació es pot veure l'exemple detallat a l' Annex A: Propagació de la informació en el disseny de la solució dinàmica.

## 4. DESENVOLUPAMENT I IMPLEMENTACIÓ

A continuació es detallarà com es decideix si es fa o no interacció i quin tipus d'interacció es fa en cadascun dels dos modes possibles.

### 4.1. Solució estàtica

Els elements que es tenen en compte a l'hora de decidir-ho són els que es mostren a continuació. S'utilitzarà el següent exemple per la definició dels elements: *L'usuari A crea la publicació Z. El programa ha de decidir si l'usuari B interacciona o no.*

- Degrees: nombre de connexions que tenen els nodes amics. Si una persona té molts amics, és probable que no vegi totes les publicacions, només aquells amb qui tingui més afinitat.

Per calcular-lo es normalitzarà aquest valor. Com per a interaccionar és més òptim que l'usuari tingui poques connexions, el degree serà 1-la normalització:

$$Degree = 1 - \frac{Degree_{userB} - Degree_{min}}{Degree_{max} - Degree_{min}}$$

- Weights: pes que té l'enllaç entre l'usuari i els seus amics, com més alt més forta és la connexió (va entre 0 i 1).
- Quality: Qualitat de les publicacions. Les de gran qualitat són més propícies a obtenir likes. La qualitat serà generada prenent un valor aleatori entre 0 i 1.
- Relation: Relació entre el topic de la publicació i els interessos de l'usuari. És molt més probable que una persona doni like a una publicació que li interessi el contingut. Si el topic de la publicació Z és un dels interessos de l'usuari B, aleshores Relation serà 1. En cas contrari, Relation serà 0.
- Age: Edat de les persones. Persones amb edat avançada tendeixen a fixar-se més en el contingut i la seva qualitat que en la persona que el publica. Per contra, una persona jove tendeix a fixar-se més en l'usuari que publica els continguts.
- Type: Tipus de publicació. Les imatges són les publicacions amb més interaccions. Al mateix temps, els vídeos acostumen a tenir més interaccions que els texts. Es defineix de la següent manera:

Type of publication	Type
Image	1
Video	0,5
Text	0

L'algorisme que s'encarrega de prendre la decisió fa el següent: per cada publicació que es creï i per cada node amb el qual estigui connectat el node creador, es calcularà una puntuació entre 0 i 1 donant els següents pesos als diferents camps:

Rate	Age		
	18-25	26-45	46-85
Degree	0,15	0,15	0,15
Weight	0,35	0,25	0,2
Quality	0,05	0,1	0,14
Type	0,1	0,05	0,01
Relation topic-Interest	0,35	0,45	0,5

Per exemple, en el cas que li arribi a un usuari de 22 anys, calcularà l'score de la següent manera:

$$\textit{score} = 0,15 * \textit{degree} + 0,35 * \textit{weight} + 0,05 * \textit{quality} + 0,1 * \textit{type} + 0,35 * \textit{relation}$$

La probabilitat que l'usuari A faci like a la publicació 1 és igual la *score* obtinguda. La probabilitat que també faci un comentari és *score*-0,2. Tot i això hi ha un score mínim necessari per tal de que l'usuari interactuï. Per defecte és 0,3.

La probabilitat que l'usuari no interactui és 1-*score*.

El programa amb els paràmetres per defecte triga entre 1 i 3 segons.

A l' Annex B: *Pseudo-codi de la solució estàtica* podem trobar el pseudo-codi de la solució estàtica.

## 4.2. Solució dinàmica

A la solució dinàmica ens trobarem amb les següents opcions:

1. Usuari no interacciona.
2. Usuari interacciona:
  - a. Usuari dona like a una publicació.
  - b. Usuari dona like i comenta una publicació.
  - c. Usuari dona like, comenta i comparteix (share) una publicació.

Per a simplificar els processos, la xarxa social creada tindrà les següents normes:

- Si l'usuari no interacciona en distància 1 amb una publicació, pot tenir (o no) qualsevol interacció del tipus 2 en qualsevol distància posterior a aquesta. Cada cop, però, tindrà menor probabilitat de fer-ho.
- Si l'usuari ha interaccionat amb una publicació en qualsevol de les distàncies, ja no interaccionarà més amb aquesta publicació. És a dir, si l'usuari ha fet una interacció del tipus 2.a (like) en distància 1 d'interacció, ja no podrà fer les interaccions 2.b ni

2.c en distàncies posteriors. És a dir, no podrà comentar o compartir la publicació a no ser que ho hagi fet en primera instància.

- Si l'usuari ha interaccionat amb una publicació, aquesta interacció no es podrà desfer.

Arribats a aquest punt, el que queda és definir com i quan es produiran (o no) les interaccions tal com hem fet amb la versió estàtica. Per fer-ho un algorisme principal ens calcularà la puntuació de manera diferent segons la distància a la que ens trobem.

## 1. Distància 1

Semblant a l'algorisme de la solució estàtica, però incorporant l'opció d'etiquetar usuaris i la possibilitat de compartir la publicació.

Els nous elements que es tenen en compte a l'hora de decidir l'score són els que es mostren a continuació. S'utilitzarà el següent exemple per la definició dels elements: *L'usuari A crea la publicació Z. El programa ha de decidir si l'usuari B interacciona o no.*

- Relation old publications-user: Analitza les interaccions que ha fet l'usuari B amb les antigues publicacions fetes per l'usuari A. Si l'usuari B ha interaccionat amb la majoria de publicacions fetes per l'usuari A en el passat, és probable que interacció amb les següents de la mateixa manera.

$$\text{relation\_oldpub\_user} = \frac{\text{num}_{\text{interactions}}}{\text{total}_{\text{publications}_{\text{userA}}}}$$

on num\_interactions és el nombre d'interaccions fetes per l'usuari B amb les publicacions passades de l'usuari A.

- Relation topic – old publications: Analitza les interaccions que ha fet l'usuari B amb publicacions anteriors que tinguin el mateix topic que la publicació 1. Si un usuari està constantment interaccionant amb el topic Z de la publicació 1, és probable que vulgui interaccionar amb aquesta.

$$\text{relation\_topic\_oldpub} = \frac{\text{num}_{\text{interactions}_{\text{topicZ}}}}{\text{total}_{\text{possible}_{\text{interactions}_{\text{topicZ}}}}}$$

on num\_interactions\_topic és el nombre total d'interaccions que ha fet l'usuari B amb les publicacions del topic Z i total\_possible\_interactions\_topicZ és el total de publicacions amb topic Z amb les que podria haver-hi interaccionat l'usuari B. Destacar que no és el mateix que el nombre total de publicacions amb topic Z, ja no totes les publicacions li arribaran a l'usuari per poder tenir la possibilitat d'interaccionar.

També es tindran en compte les variables  $\beta$ ,  $\mu$  i  $\alpha$ . Aquestes aniran decreixent amb el pas del temps. L'objectiu d'aquestes és canviar el pes d'alguns dels elements. Inicialment, a  $t=0$ , no es té informació sobre les interaccions amb publicacions passades, per tant, s'ha de calcular la probabilitat d'interacció de la mateixa manera que amb la solució sense

historial. Per contra, a mesura que va passant el temps, es poden tenir en compte les interaccions anteriors, que ens permetran ser més precisos per calcular les probabilitats.

Per exemple, pel que fa a l'element Relation topic - old publications no es tindrà en compte inicialment. En el seu lloc es valorarà únicament la relació entre el topic de la publicació i els interessos que té l'usuari B. Però a mesura que vagi passant el temps es tindran en compte la relació entre el topic de la publicació actual, i el topic de les publicacions amb les quals ha interactuat previament l'usuari B.

Per tant, de la mateixa manera que amb la solució estàtica, el que farem serà definir un percentatge d'influència a cadascun dels elements. Aquest cop serà una mica més complex. En la següent taula es mostren els pesos que se'ls hi dona a cada element:

Rate	Age		
	18-25	26-45	46-85
Degree	$\beta (=0,15 \text{ at } t=0)$	$\beta (=0,15 \text{ at } t=0)$	$\beta (=0,15 \text{ at } t=0)$
Weight	$\mu (=0,35 \text{ at } t=0)$	$\mu (=0,25 \text{ at } t=0)$	$\mu (=0,2 \text{ at } t=0)$
Relation old publications- user	$0.5-\beta-\mu$	$0.4-\beta-\mu$	$0.35-\beta-\mu$
Quality	0,05	0,1	0,14
Type	0,1	0,05	0,01
Relation topic- Interest	$\alpha (=0,35 \text{ at } t=0)$	$\alpha (=0,45 \text{ at } t=0)$	$\alpha (=0,5 \text{ at } t=0)$
Relation topic -old publications	$0,35-\alpha$	$0,45-\alpha$	$0,5-\alpha$
Tagged (extra)	0,05	0,05	0,05

Per exemple, per a un usuari de 22 anys calcularem l'score de la següent manera:

$$\begin{aligned} score = & \beta * degree + \mu * weight + (0.5 - \beta - \mu) * relation\_oldpub\_user + 0.05 \\ & * quality + 0.1 * type + \alpha * relation\_topic\_interests + (0.35 - \alpha) \\ & * relation\_topic\_oldpub + tagged \end{aligned}$$

Per a les distàncies 2 i 3 podem trobar l'explicació, les taules de com calcular la puntuació i exemples l' Annex C: *Càlcul de les distàncies 2 i 3 en la versió dinàmica*.

A l' Annex D: *Pseudo-codi de la solució dinàmica* trobem el pseudo-codi.

A l' Annex E: *Propagació de la informació amb la versió dinàmica*. trobem un exemple complert de la propagació de la informació, fent tots els càlculs d'scores anteriorment en un graf.

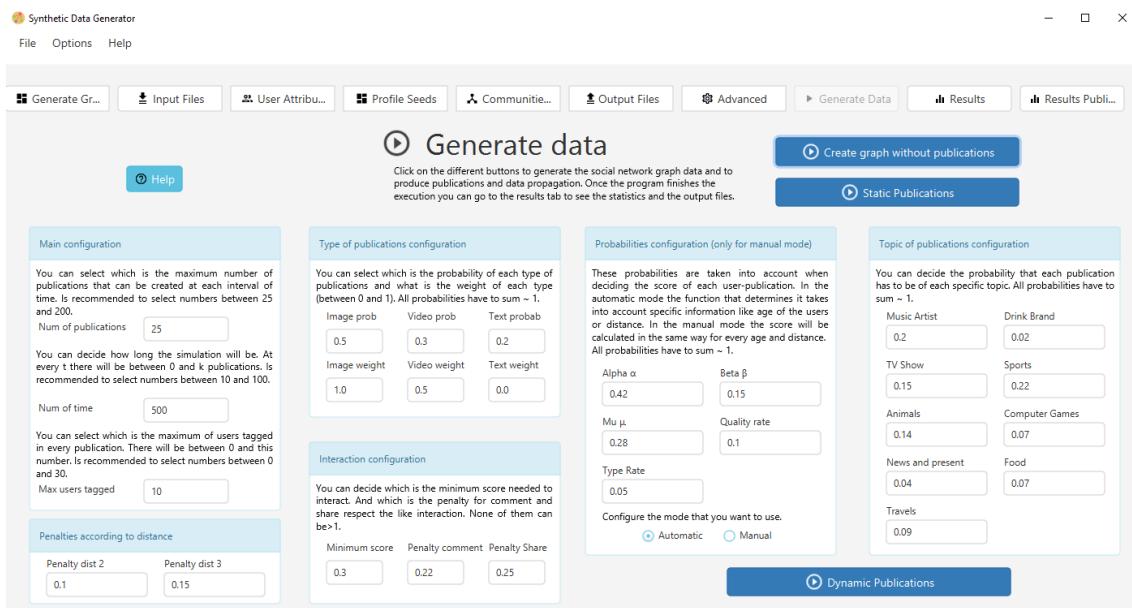
El programa amb els paràmetres per defecte triga aproximadament 24 minuts ja que hi ha un alt cost computacional. Baixant el nombre de publicacions rebaixem el temps d'execució considerablement.

### 4.3. Pantalles de la interfície d'usuari

Quan s'executa el programa en Java s'obre la interfície d'usuari. Aquesta permet navegar per diferents pestanyes per a configurar el model, crear el graf, produir la propagació de les dades i veure un resum dels resultats i estadístiques.

A l' Annex I: *Explicació de les pantalles de la interfície d'usuari* podem veure una explicació detallada de cadascuna de les pantalles. S'explica per a què serveix cada una i com configurar correctament el programa.

Les dues que he creat jo, corresponents a la propagació de la informació i al resum de resultats de les publicacions són les següents:



#### 4.3.1. Interfície de l'usuari - Generate Data

La pestanya *Generate Data* ens permet de manera interactiva configurar els diferents paràmetres que configuren el model i els algorismes. Per a cada valor hi ha una explicació de per a què serveix.

Com a configuració principal, es pot modificar el nombre de publicacions màximes que es poden fer a cada instant de temps, les unitats de temps que es vol que passin en el programa i el màxim d'usuaris etiquetats per publicació.

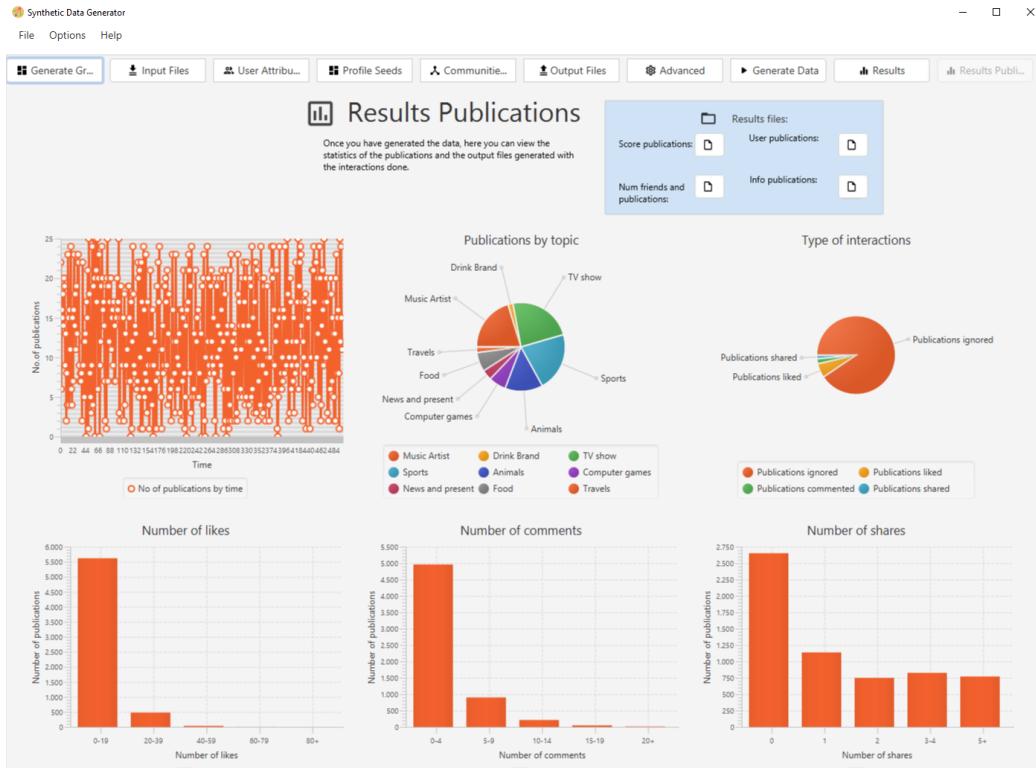
A continuació, es pot configurar les penalitzacions per a les puntuacions de distàncies 2 i 3. També, permet configurar la probabilitat per a cada publicació que sigui imatge, vídeo o text i el pes que pren cadascun dels tipus de publicació en el moment de calcular l'*'score'*.

Es permet la configuració per a decidir el tipus d'interacció un cop està l'*'score'* calculada, decidint la mínima puntuació necessària per a poder interaccionar, i les penalitzacions per comentar i compartir la publicació respecte a l'acció de fer like.

També, es pot configurar els pesos que s'assignen als diferents paràmetres utilitzats per a calcular l'*'score*. Només serveix per al mode manual i per a totes les distàncies i rangs d'edat s'usaran aquests valors. Per contra, en el mode automàtic (recomanat), com s'ha explicat a la secció anterior, per a cada distància i per a cada rang d'edat s'assignen valors diferents als paràmetres que permet que la simulació s'adapti més a la realitat.

Per últim, podem configurar la probabilitat que té cada publicació de ser d'una temàtica o d'un altre. Recordar que un cop s'assigna la temàtica a una publicació se li assigna una descripció d'aquella temàtica de manera aleatòria.

L'altra pantalla és la següent:



#### 4.3.2. Interfície de l'usuari - Results Publications

Per una banda, ens permet descarregar els fitxers de sortida més importants que s'han produït respecte les publicacions.

Per l'altra banda, ens permet veure les publicacions que s'han produït a cada instant de temps, la distribució de les diferents temàtiques de les publicacions, els tipus d'interaccions i el número de likes, comentaris i shares que s'han produït durant la simulació. Important destacar que aquesta pantalla només es genera quan s'executa la solució dinàmica.

## 5. VALIDACIÓ DE LA SOLUCIÓ

### 5.1. Estadístiques de xarxes socials reals

*Medici* és un simulador de xarxes socials que no es basa en l'exemple d'una xarxa social real concreta. És a dir, segons els paràmetres que indiquem la podríem definir com una barreja de les xarxes socials més utilitzades i ens pot servir per fer simulacions i extreure resultats per a qualsevol d'elles.

Amb les xarxes que més es podria identificar per la seva similitud respecte a continguts que es publiquen i interaccions possibles són: Facebook, Instagram i Twitter.

Un aspecte clau per analitzar les xarxes socials és el *Engagement Rate*. Aquesta mètrica és un percentatge que permet fer un seguiment de la implicació activa del públic en els continguts d'un usuari o marca. En la xarxa social *Medici* qualsevol mena d'interacció serà considerada per al seu càlcul. Per definició [13] es calcula de la següent manera:

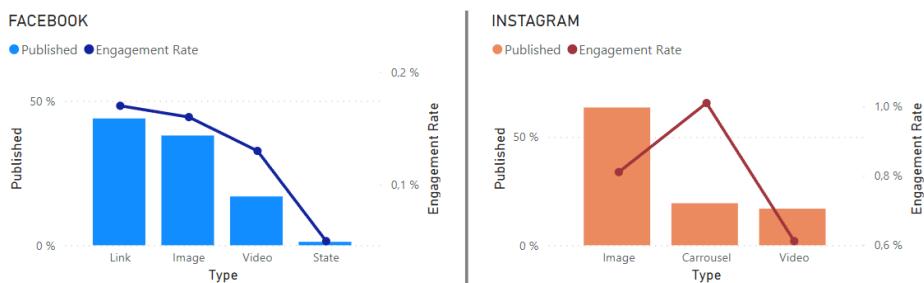
$$ER = \frac{\text{Interaccions totals en una publicació}}{\text{Num. persones a les que ha arribat la publicació}} * 100 [\%]$$

L'*Influencer MarketingHub* [15] va realitzar un estudi aquest any analitzant les principals xarxes socials i els seus continguts. En aquest ens podem trobar amb informació molt detallada, que ens permet extreure les següents dades:

Els tipus de continguts més compartits pels usuaris a Facebook al 2021 van ser els Links amb un 43,9%. Seguit de molt a prop per les imatges amb un 38%. Per últim tenim els vídeos amb un 16,9% i les actualitzacions d'estat amb un 1,2%.

Les grans marques van obtenir una mitjana de 0,08% d'Engagement Rate per publicació, que desglossat per tipus de publicació ens queda de la següent manera: Actualitzacions d'estat 0,17%, fotos 0,16%, vídeos 0,13% i Links 0,05%.

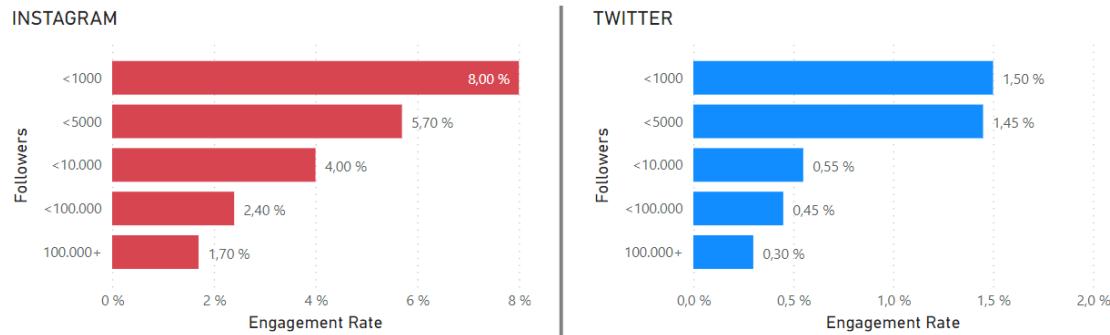
Pel que fa a Instagram, el 63,6% de publicacions van ser fotos, el 16,9% van ser vídeos i el 19,4% van ser publicacions d'històries, amb un 0,81%, un 0,61% i un 1,01% d'Engagement Rate mig respectivament en comptes amb un gran nombre d'amistats i seguidors.



5.1.1. Publicacions i Engagement Rate Facebook i Instagram

Cal tenir en compte que en el nostre simulador en la gran majoria de casos ens trobarem amb xarxes socials amb pocs usuaris, donat que el temps d'execució és limitat i incloure un nombre elevat d'usuaris fa que el programa trigui molt. Es per això que hem de mirar estadístiques on se'ns mostri l'engagement en usuaris que no tenen tantes connexions.

La mateixa plataforma *Influencer MarketingHub* [16] ens mostra una comparativa per a Instagram i Twitter comparant segons els seguidors (amistats) que tenen els usuaris. Ens trobem el següent:



5.1.2. Engagement Rate per seguidors Instagram i Twitter

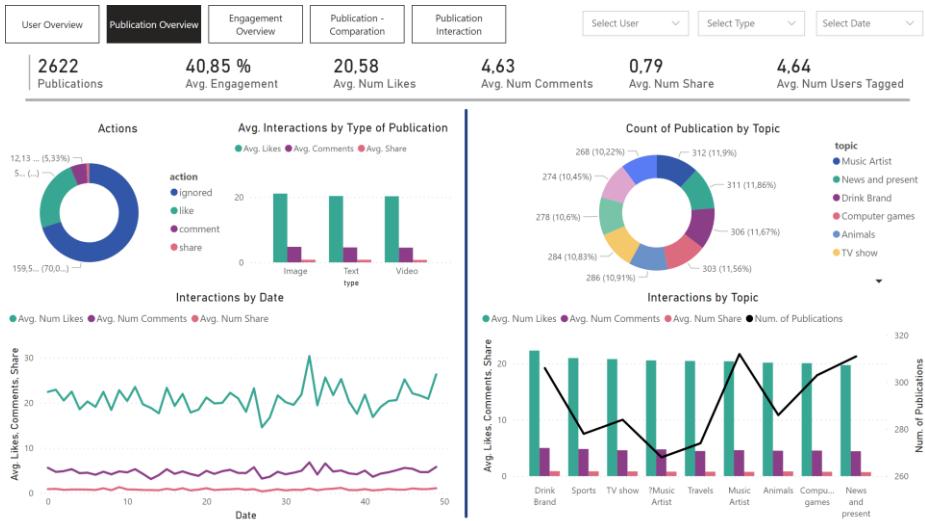
Podem veure que els casos amb menys de 1.000 seguidors augmenta significativament el Engagement Rate, arribant al 8% a l'Instagram i al 1,50% al Twitter.

A l'estudi del 2022 *Social Media Industry Benchmarks* [17] ens trobem la diferència d'Engagement Rate en funció de la temàtica de la publicació, però és molt variable segons la xarxa social. Per exemple en comptes amb molts seguidors a Instagram les publicacions sobre viatges tenen un 1,51% d'Engagement sent la temàtica amb més interaccions i a Twitter és la novena temàtica més compartida amb un 0,06%.

## 5.2. Visualització de dades amb Power BI

A més de les pantalles de resultats que es troben a *Medici*, per a tal de poder analitzar els resultats de la simulació amb més profunditat, he desenvolupat un dashboard interactiu amb PowerBI. Aquest permet consultar de forma interactiva tota la informació rellevant tant dels usuaris com de les publicacions que s'ha generat durant la simulació.

Aquesta és una de les pantalles:



#### 5.2.1. Publications Overview Power BI

A l'Annex F: *Taulell Power BI* es poden veure captures de totes les pantalles que formen el panell i explicacions detallades que mostra cadascun i com utilitzar-lo. En grans trets aquesta és la informació que ens trobem:

Permet veure el graf generat, així com informació detallada dels usuaris de la xarxa. Es troba informació detallada de les publicacions creades i gràfics on es separa segons el tipus, la temàtica o la comunitat de l'usuari que la creat. Permet visualitzar l'engagement rate segons varíes dimensions. Es pot fer comparacions d'usuaris i veure quin és el perfil de persona que està interaccionant amb cada publicació. En cadascuna de les pantalles podem fer filtres d'usuaris, publicacions, dates, tipus o temàtiques.

Un avantatge molt important, és que clicant un únic botó podem actualitzar les bases de dades que s'han generat en la nova simulació. Per tant, podrem veure de manera ràpida i senzilla tota la informació actualitzada.

### 5.3. Estadístiques Medici

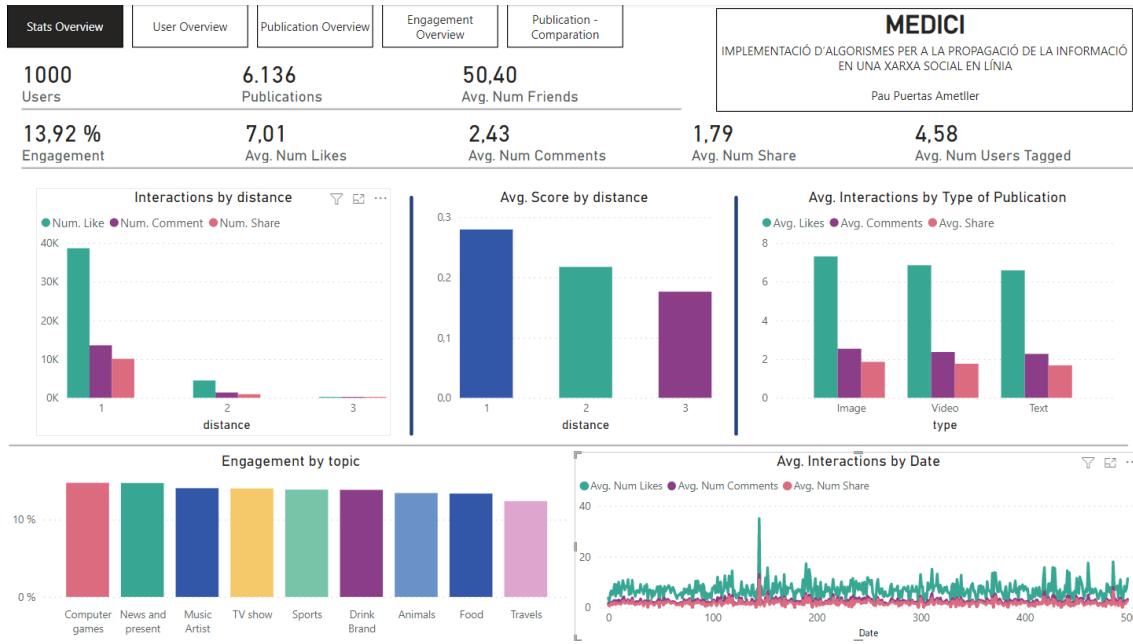
Per tal de validar els resultats i que tota la informació s'està generant i propagant correctament he fet varíes execucions amb diferents paràmetres.

#### a) Execució amb els paràmetres per defecte

En l'execució amb els paràmetres per defecte:

Num nodes	1.000
Num Edges	30.000
Max publications by time	25
Num time	500

Obtenim els següents resultats:

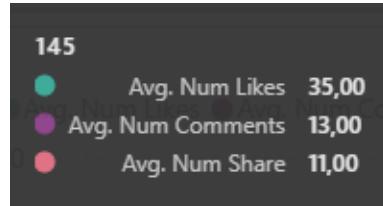


### 5.3.1. Execució amb els paràmetres per defecte

S'obté un Engagement del 13,92% on els usuaris tenen una mitjana d'amics de 50,40 usuaris. Per a cada publicació hi ha de mitjana 7,01 likes, 2,43 comentaris i 1,79 shares. Podem veure com la gran majoria d'interaccions es produeixen a distància 1, és a dir, són els usuaris que estan directament connectats amb l'usuari que penja la publicació els més interessats. Tot i això també trobem interaccions a distància 2 i en menor mesura a distància 3.

Comprovem que les imatges són les publicacions que tenen més interaccions mitjanes, seguides pels vídeos. Si ens fixem en les temàtiques, veiem que les publicacions de videojocs i notícies interessen lleugerament més que les altres.

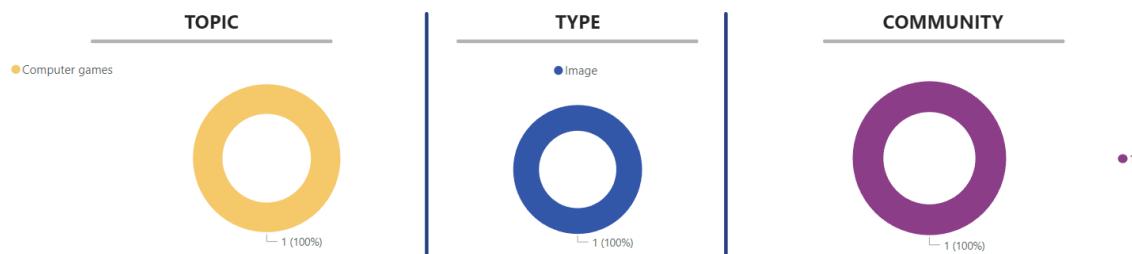
Si analitzem més en concret aquesta simulació amb el Power Bi, trobem que en l'instant de temps 145 hi ha hagut una forta pujada de la mitjana de likes, comentaris i shares:



### 5.3.2. Interaccions a l'instant 145

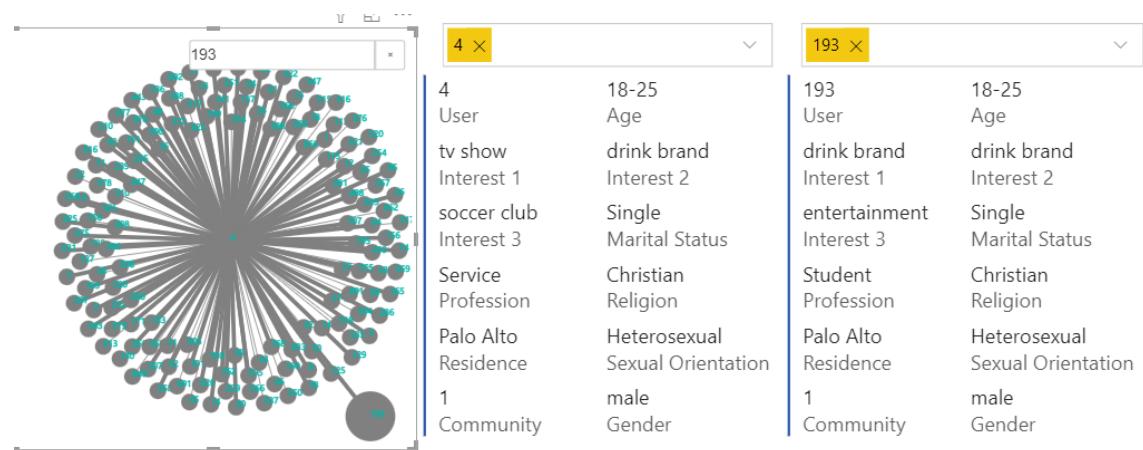
Veiem que en aquest instant únicament s'ha produït 1 publicació, la qual ha obtingut 69,45% d'engagement.

Aquesta és la 1916 i es tracta d'una imatge, amb temàtica videojocs i feta per un usuari de la comunitat 1.



#### 5.3.3. Topic, type and community of publication 1916

L'usuari creador de la publicació és el 4, el qual té 145 amics, pràcticament triplicant la mitjana. Un dels seus amics, l'usuari 193, ha obtingut un score de 0,36 envers aquesta publicació a distància 1 i l'ha compartit. Podem veure a continuació les principals característiques del creador de la publicació i de l'usuari que ha fet share:



#### 5.3.4. Característiques dels dos usuaris

Veiem com els dos són de la mateixa edat, tenen uns interessos semblants, els dos estan solters, segueixen la mateixa religió i viuen a la mateixa ciutat. A més són de la mateixa comunitat.

El fet que l'usuari 193 i altres usuaris comparteixin la publicació la fa arribar a nous usuaris. Hi ha 5 usuaris que han interaccionat. Deixar clar que es mostra únicament la interacció de major pes en cada cas. És a dir, el share de l'usuari 529 inclou share+comment+like:

Publication	Distance	User	Interaction	Score
1916	2	490	comment	0,32
1916	2	833	like	0,31
1916	2	523	like	0,31
1916	2	529	share	0,31
1916	2	462	like	0,30

#### 5.3.5. Interaccions a distàncies 2 i 3

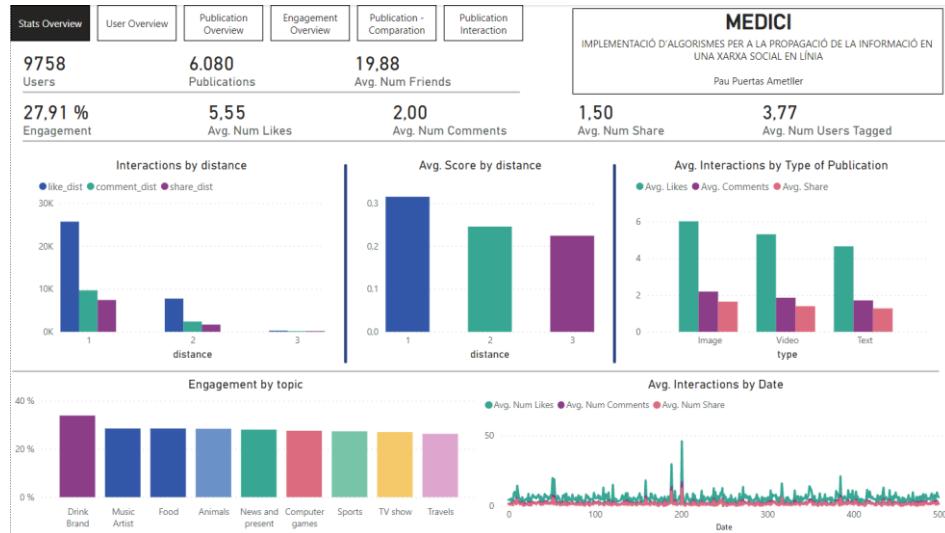
Veiem doncs com el programa pren sentit i gràcies a la flexibilitat del Power BI, podem anar analitzar cada apartat per seguir la propagació de la informació.

### b) Execució amb una xarxa més gran

En una altra execució amb més nodes i els següents paràmetres:

Num nodes	10.000
Num Edges	100.000
Max publications by time	25
Num time	500

Obtenim els següents resultats:



#### 5.3.6. Execució amb una xarxa més gran

En aquest cas la mitjana d'amics és de només 19,88 per usuaris. Això provoca que els contactes siguin més fiels i l'engagement augmenti fins al 27,91%. Per tant, veiem com amb pràcticament les mateixes publicacions, les interaccions per temàtica augmenten pràcticament el doble. En aquest cas, les publicacions sobre marques de beguda han obtingut un engagement per sobre de la resta. La mitjana d'interaccions amb el pas del temps es manté estable, excepte en alguns instants de temps concrets on creix considerablement.

Altres execucions combinant diversos paràmetres les trobem a l' *Annex F: Més execucions del programa*. S'expliquen les diferències entre les diferents execucions i com afecten els paràmetres a aquestes.

### c) Limitacions del programa

Ens trobem amb dues grans limitacions:

Amb el programa *Medici* no podem aplicar la simulació en xarxes amb milions de nodes i altament connectats. Això ens serviria per poder relacionar els resultats de les nostres simulacions amb els resultats estudiats de les xarxes socials més utilitzades amb un gran nombre d'amistats i seguidors. Tot i això, veiem que a mesura que augmentem el nombre d'amistats que tenen els usuaris, es redueix considerablement l'engagement, tal com hem vist que succedia a Instagram i Twitter en l'apartat anterior.

Una altra limitació important és que de moment l'usuari s'ha d'adaptar a les temàtiques que venen generades pel programa. Tot i això, s'ha intentat que les temàtiques proposades abasteixin pràcticament la totalitat de temes que es tracten a les xarxes socials. A més cada descripció s'assigna únicament a una temàtica i a la realitat una mateixa publicació podria tractar de diversos temes.

Finalment, i com no podia ser d'una altra manera, és una simulació que s'ha desenvolupat amb un temps limitat, i caldria dedicar un període de temps molt més ampli per a tal d'assignar els paràmetres a la perfecció i afegir-ne d'altres per a fer-la més exacte.

Futures implementacions del programa podrien ser incorporar com a fitxers d'entrada publicacions de xarxes socials reals (com podrien ser tweets), que el programa mitjançant intel·ligència artificial reconegui de quin/quins temes tracta cada publicació i que siguin aquestes les que es propaguin per la xarxa.

També, adaptar la creació de la xarxa per a poder fer-la amb un gran nombre de nodes i connexions seria una implementació molt vàlida per a poder millorar amb la simulació.

## 6. CONCLUSIONS

A l'inici es plantejava la creació d'una simulació per a xarxes socials com a un projecte ambiciós. Finalment, podem assegurar que s'ha aconseguit l'objectiu principal: la implementació d'algorismes que permeten propagació de la informació.

Inicialment, vaig fer un estudi general de les xarxes socials, de la seva dimensió i del que permeten a la societat. A continuació i després d'analitzar amb quina freqüència interactuen els usuaris amb les publicacions a les xarxes i en què es basen per a fer-ho, vaig començar a plantejar el disseny de la solució. Es plantejava com complex, donades la gran quantitat de factors que intervenen per a què un usuari decideixi interaccionar. És per això que vaig decidir fer una primera solució bàsica i estàtica que permetés fer el salt a la solució dinàmica adaptada a la realitat.

En aquesta primera solució només es tenia en compte la informació que era causada pel programa en l'instant de produir-se la creació de la xarxa social. És a dir, les característiques i atributs d'aquests. Per tant, els resultats eren molt precaris i distànciats de les xarxes socials reals que tots utilitzem avui en dia.

En la solució dinàmica, vaig usar l'esquelet de la solució inicial per a poder produir la correcta propagació de la informació. Els algorismes que he creat tenen en compte una gran quantitat de factors que plegats permeten obtenir resultats semblants als que ens podríem trobar a la realitat.

El llenguatge de programació ha estat una gran barrera per a mi donada la inexperiència que tenia amb aquest. Però finalment s'ha pogut fer el salt del disseny a la solució donant-nos els resultats explcats.

Finalment, el desenvolupament del *dashboard* amb Power BI ha permès donar-li forma a tot el projecte visualitzant la totalitat de la informació generada. Gràcies a la interacció ens permet entendre que és el que s'ha produït en cada moment i perquè i, analitzar els resultats per poder extreure les conclusions pertinents.

Tot els fitxers generats, el projecte fet en Java/JavaFX i el programa fet amb Power BI es poden trobar al següent enllaç: <https://github.com/dnettle/MEDICIP>.

## 7. BIBLIOGRAFIA

- [1] Chaffey, D. (2022, 31 May). Global social media statistics research summary 2022. Smart Insights. <https://www.smartinsights.com/social-media-marketing/social-media-strategy/new-global-social-media-research/>
- [2] Kemp, S. (2022, 4 May). Digital 2022: Global Overview Report. DataReportal – Global Digital Insights. <https://datareportal.com/reports/digital-2022-global-overview-report>
- [3] Ofcom. (2021, April). Children and parents: Media use and attitudes report 2020/21. [https://www.ofcom.org.uk/\\_data/assets/pdf\\_file/0025/217825/children-and-parents-media-use-and-attitudes-report-2020-21.pdf](https://www.ofcom.org.uk/_data/assets/pdf_file/0025/217825/children-and-parents-media-use-and-attitudes-report-2020-21.pdf)
- [4] Kakkar, G. (2022, 14 January). What Are The Different Types Of Social Media? Digital Vidya. <https://www.digitalvidya.com/blog/types-of-social-media/>
- [5] Lin, Y. Y. (2020, 21 January). The 6 Types of Social Media with Examples. SEOPressor - Best SEO Wordpress Plugin. <https://seopressor.com/social-media-marketing/types-of-social-media/>
- [6] F.N. Sotkman. (2002, 2 November). ScienceDirect. Networks: Social. <https://www.sciencedirect.com/science/article/pii/B0080430767019343>
- [7] Estivill-Castro, V., & Nettleton, D. F. (2015, August). Can on-line social network users trust that what they designated as confidential data remains so?. In 2015 IEEE Trustcom/BigDataSE/ISPA (Vol. 1, pp. 966-973). IEEE.
- [8] Nettleton, D. F. (2016). A synthetic data generator for online social network graphs. Social Network Analysis and Mining, 6(1), 1-33.
- [9] Nettleton, D. F., & Nettleton, S. (2021, September). MEDICI: A simple to use synthetic social network data generator. In International Conference on Modeling Decisions for Artificial Intelligence (pp. 273-285). Springer, Cham.
- [10] Torra, V. T. (2021, 27 septiembre). Modeling Decisions for Artificial Intelligence. Google Books. <https://books.google.es/books?id=IOFDEAAAQBAJ&pg=PA273&lpg=PA273&dq=works+that+use+medici++synthetic+social+network+data+generator&source=bl&ots=1SAiuQXhds&sig=ACfU3U1WDKqXpwngHuPHBOR0nBEgLmzQhg&hl=ca&sa=X&ved=2ahUKEwjZo5627bT4AhUkzoUKHTU0BEYQ6AF6BAGYEAM#v=onepage&q=medici&f=false>

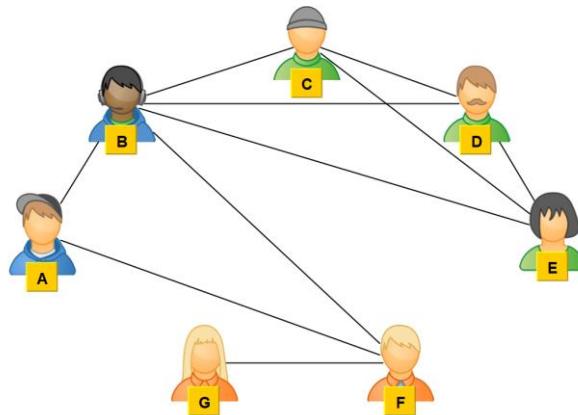
- [11] Ade-Ojo, J. (2021, 27 December). The Extreme Power of Long Tailed Distributions - Towards Data Science. Medium. <https://towardsdatascience.com/the-power-of-long-tailed-distributions-bd46f8856039>
- [12] Wikipedia contributors. (2022, 27 febrero). Chris Anderson. Viquipèdia, l'enclopèdia lliure. [https://ca.wikipedia.org/wiki/Chris\\_Anderson](https://ca.wikipedia.org/wiki/Chris_Anderson)
- [13] GeeksforGeeks. (2021, 31 May). Graph measurements: length, distance, diameter, eccentricity, radius, center. <https://www.geeksforgeeks.org/graph-measurements-length-distance-diameter-eccentricity-radius-center/>
- [14] A. (2019, 23 April). The Six Degrees of Separation Theory. Exploring Your Mind. <https://exploringyourmind.com/the-six-degrees-of-separation-theory/>
- [15] Corporate Finance Institute. (2022, 29 abril). Engagement Rate. <https://corporatefinanceinstitute.com/resources/knowledge/e-commerce-saas/engagement-rate/>
- [16] Geyser, W. (2022, 4 febrero). Social Media Marketing Benchmark Report 2022. Influencer Marketing Hub. <https://influencermarketinghub.com/social-media-marketing-benchmark-report/>
- [17] Influencer Marketing Hub. (2018, 29 agosto). Micro-influencers vs Celebrities. <https://influencermarketinghub.com/micro-influencers-vs-celebrities/>

## 8. ANNEXOS

### Annex A: Propagació de la informació en el disseny de la solució dinàmica

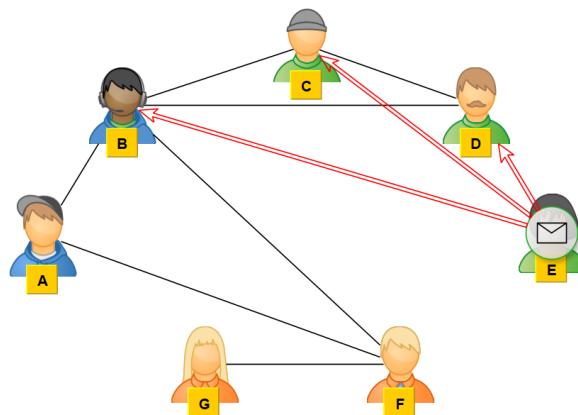
A continuació es mostra un exemple de com es produeix la propagació de la informació en el disseny de la solució dinàmica. Per simplificar-lo, cada cop que un usuari decideixi interaccionar amb una publicació farà like + comentari + compartir:

1. Ens trobem en una xarxa amb 7 usuaris. Existeix un enllaç entre dos usuaris quan estan connectats. Veiem com el node B seria considerat un influencer ja que està connectat amb 5 dels usuaris de la xarxa. Altres usuaris com l'A i el G tenen molt poques connexions.



A. Nodes i enllaços

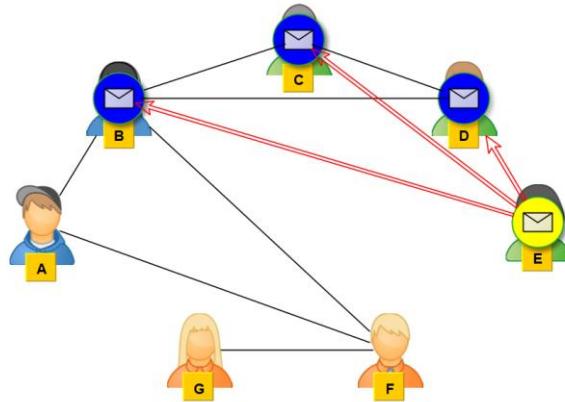
2. L'usuari E fa una publicació. Tot l'exemple tractarà sobre aquesta. Aquesta és propagada cap els nodes amb els que està connectat: B, C i D.



A. Nova publicació

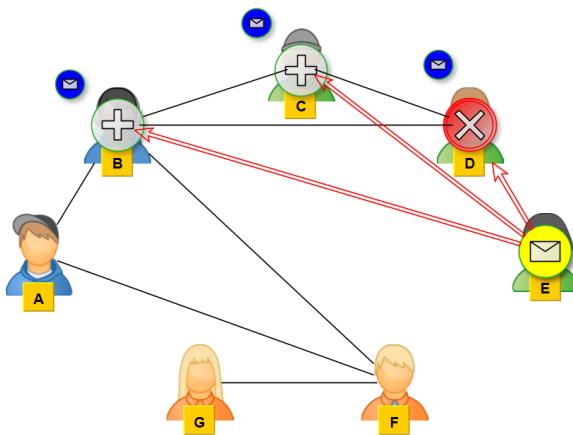
3. Els nodes B, C i D reben la publicació.

La representem en color blau per indicar que la possible interacció amb la publicació és de distància 1.



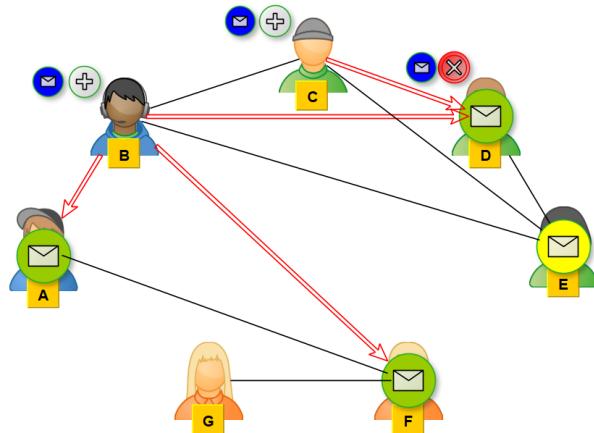
A. Rebuda de publicacions a distància 1

4. Els usuaris decideixen si interaccionar o no amb la publicació. Els usuaris B i C interaccionen, l'usuari D no ho fa.



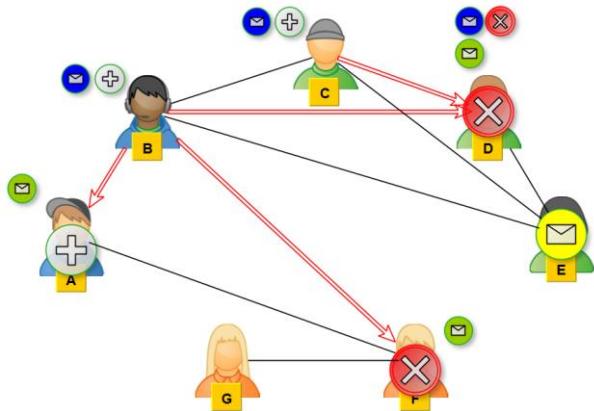
A. Interaccions a distància 1

5. Els usuaris que han interaccionat, propaguen la publicació amb els usuaris que tenen connexió. Si un usuari ha interaccionat anteriorment o és el creador de la publicació no rep més cops la publicació. Podem veure com l'usuari B, al ser un influencer, fa arribar la publicació a un gran nombre d'usuaris.  
S'indica en color verd que la possible interacció que es produueixi entre l'usuari i la publicació serà de distància 2.



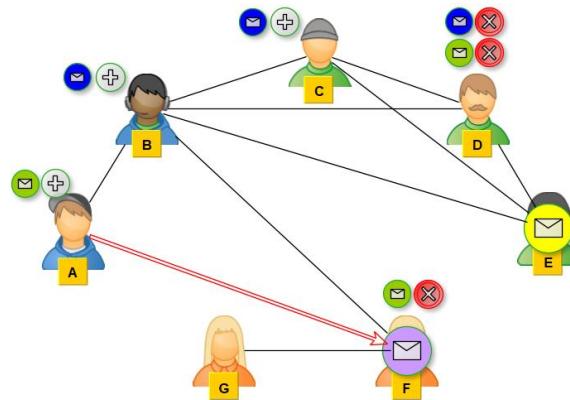
A. Rebuda de publicacions a distància 2

6. Els usuaris decideixen si interaccionen amb la publicació. En aquest cas només ho fa l'usuari A.



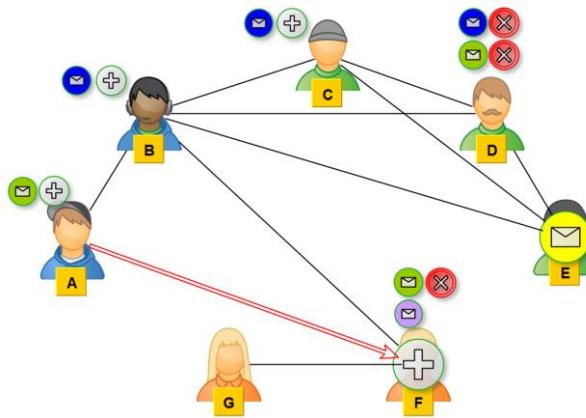
A. Interaccions a distància 2

7. L'usuari A propaga la publicació cap als usuaris amb els que està connectat. S'indica en color lila que la possible interacció que es produueixi entre l'usuari F i la publicació serà de distància 3.



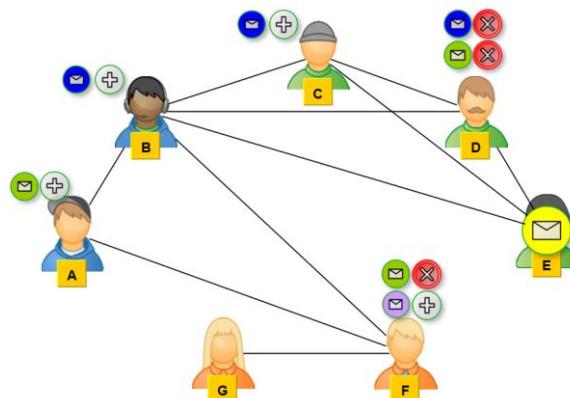
A. Rebuda de publicacions a distància 3

8. L'usuari F interacciona amb la publicació.



A. Interaccions a distància 3

9. Suposant que la distància màxima és 3 en aquest exemple, la propagació d'aquesta publicació hauria finalitzat. Veiem com finalment la publicació ha aconseguit les interaccions dels usuaris A, B, C i F. Tot i que alguns d'aquests (A i F) no estan connectats amb l'usuari que crea la publicació (E), aquests han acabat interaccionant igualment, donat que l'han rebut d'altres usuaris que han interaccionat compartint la publicació .



A. Graf final

## Annex B: Pseudo-codi de la solució estàtica

A continuació es mostrà el pseudo-codi de la solució estàtica:

---

**Algorithm 1:** genPublicationsSV

---

```
for user in users do
    Publications are generated with the probabilities of the long tail function;
for publication in publications do
    for friend in user.friends do
        score := scoreFunction(user, publication);
        interaction := interactionFunction(score);
        if interaction == 2 then
            user do: like and comment
        else if interaction == 1 then
            user do: like
        else
            user do: nothing
```

---

---

**Algorithm 2:** scoreFunction

---

```
input: user, publication;
output: float between 0 and 1;
if user.age ≤ 25 then
    degree_rate := 0.15;
    weight_rate := 0.35;
    quality_rate := 0.05;
    type_rate := 0.1;
    relation_rate := 0.35;
else if user.age ≥ 25 and user.age ≤ 45 then
    degree_rate := 0.15;
    weight_rate := 0.25;
    quality_rate := 0.1;
    type_rate := 0.05;
    relation_rate := 0.45;
else
    degree_rate := 0.15;
    weight_rate := 0.2;
    quality_rate := 0.14;
    type_rate := 0.01;
    relation_rate := 0.5;
degree := user.degree;
weight := weight(user, publication.user_id);
quality := publication.quality;
if publication.type == "Image" then
    type := 1;
else if publication.type == "Video" then
    type := 0.5;
else
    type := 0.0;
if user.like1 or user.like2 or user.like3 == publication.topic then
    relation := 1;
else
    relation := 0;
score = degree_rate*degree + weight_rate*weight + quality_rate*quality + type_rate*type
+ relation_rate*relation;
return score;
```

---

---

**Algorithm 3:** interactionFunction

---

```
input: score;
output: int that can be 0, 1 or 2;
rand_num1 := random_number between (0,1);
if score ≥ rand_num and score ≥ 0,3 then
    rand_num2 := random_number between (0,1);
    if score - 0.2 ≥ rand_num then
        return 2;
    else
        return 1;
else
    return 0;
```

---

## Annex C: Càlcul de les distàncies 2 i 3 en la versió dinàmica

### I. Distància 2

En aquest cas ens trobarem amb dues possibilitats segons si l'usuari és el primer cop que veu la publicació o si per contra ja tenia la puntuació calculada amb aquella publicació a distància 1.

En ambdós casos definirem les puntuacions sobre el següent cas:

*User A fa la publication 1 amb topic Z, user B interacciona i comparteix la publicació 1, user C rep la publicació i ha de decidir si interacciona.*

- a) L'usuari C no ha tingut l'oportunitat d'interaccionar a distància 1

En aquest cas la puntuació es calcularà de manera molt semblant a l'anterior. Tot i això s'afegeixrà una penalització ja que no és igual de probable que interactuï l'usuari C si no és amic de l'usuari A. La penalització és per defecte de 0,10.

L'únic element nou que s'afegeix és el següent:

- Relation old shared publications - user: Analitza el nombre de cops que l'usuari C ha interaccionat amb publicacions que l'usuari B ha compartit amb ell.

$$\text{relation\_oldshared\_user} = \frac{\text{interactions}_{\text{done}}}{\text{total}_{\text{possible interactions}}}$$

On  $\text{interactions}_{\text{done}}$  és el nombre d'interaccions que ha fet l'usuari C en publicacions que l'usuari B li ha compartit, i el total és el nombre d'interaccions que l'usuari B li ha compartit al C.

Algunes consideracions importants són les següents:

- No es pot tenir en compte el weight entre l'usuari A i l'usuari C, ja que si no ha tingut l'oportunitat d'interaccionar amb distància 1, vol dir que aquests dos usuaris no estan connectats. Per tant sempre és 0.
- Es suprimeix l'extra de tagged, ja que un usuari només pot etiquetar en publicacions a gent amb la que està connectada ( $\text{weight} > 0$ ).

La taula amb els pesos que se li dona a cada element és la següent:

Rate	Age		
	18-25	26-45	46-85
Degree	$\beta (=0,15 \text{ at } t=0)$	$\beta (=0,15 \text{ at } t=0)$	$\beta (=0,15 \text{ at } t=0)$
Weight_UserB_UserC	$\mu (=0,35 \text{ at } t=0)$	$\mu (=0,25 \text{ at } t=0)$	$\mu (=0,2 \text{ at } t=0)$
Relation old publications- user	$(0.5-\beta-\mu)/2$	$(0.4-\beta-\mu)/2$	$(0.35-\beta-\mu)/2$
Relation old shared publications - user	$(0.5-\beta-\mu)/2$	$(0.4-\beta-\mu)/2$	$(0.35-\beta-\mu)/2$

<b>Quality</b>	0,05	0,1	0,14
<b>Type</b>	0,1	0,05	0,01
<b>Relation topic- interests</b>	$\alpha (=0,35 \text{ at } t=0)$	$\alpha (=0,45 \text{ at } t=0)$	$\alpha (=0,5 \text{ at } t=0)$
<b>Relation topic -old publications</b>	$0,35-\alpha$	$0,45-\alpha$	$0,5-\alpha$
<b>Penalty</b>	0.10	0.10	0.10

Per exemple per a un usuari de 22 anys calcularem l'score de la següent manera:

$$\begin{aligned}
 \textit{score} = & \beta * \textit{degree} + \mu * \textit{weight} + \frac{(0.5 - \beta - \mu)}{2} * \textit{relation\_oldpub\_user} \\
 & + \frac{(0.5 - \beta - \mu)}{2} * \textit{relation\_oldshared\_user} + 0.05 * \textit{quality} + 0.1 * \textit{type} \\
 & + \alpha * \textit{relation\_topic\_interests} + (0.35 - \alpha) * \textit{relation\_topic\_oldpub} \\
 & - \textit{Penalty}
 \end{aligned}$$

- b) L'usuari C ha tingut l'oportunitat d'interaccionar amb distància 1 i no ho ha fet

Es tindrà majoritàriament en compte l'score de la distància 1 i en petita escala el weight entre l'usuari B i C i la Relation old shared publications – user.

Algunes consideracions importants són les següents:

És molt important la puntuació a distància 1 ja que la percepció de l'usuari cap a la publicació no hauria de variar gaire.

No es fa restricció d'edat, ni es fa ús dels demes elements ja que al càlcul de la puntuació a distància 1 ja s'han tingut en compte.

Els pesos als diferents elements queden així:

<b>Rate</b>	<b>Age</b>
	<b>18-85</b>
<b>Score depth 1</b>	0.7
<b>Weight_UserB_UserC</b>	0.1
<b>Relation old shared publications - user</b>	0.2
<b>Penalty</b>	0.10

Per a qualsevol usuari es calcularà la puntuació de la següent manera:

$$\textit{score} = 0.7 * \textit{score}_{d1} + 0.1 * \textit{weight} + 0.2 * \textit{relation\_oldshare\_user} - \textit{Penalty}$$

## II. Distància 3

Ens trobarem amb tres possibilitats diferents segons si l'usuari és el primer cop que es troba amb la publicació o ha tingut opció en el passat d'interactuar amb aquesta.

Utilitzarem en tots els casos el següent exemple:

*User A fa la publication 1 amb topic Z, user B interacciona i comparteix la publicació, user C rep la publicació i ha de decidir si interacciona.*

- a) L'usuari C no ha tingut l'oportunitat d'interaccionar a distància 1 i 2

Serà igual que en el cas de distància 2, però una penalització més gran. Per defecte serà del 0,15.

Rate	Age		
	18-25	26-45	46-85
<b>Degree</b>	$\beta (=0,15 \text{ at } t=0)$	$\beta (=0,15 \text{ at } t=0)$	$\beta (=0,15 \text{ at } t=0)$
<b>Weight_UserB_UserC</b>	$\mu (=0,35 \text{ at } t=0)$	$\mu (=0,25 \text{ at } t=0)$	$\mu (=0,2 \text{ at } t=0)$
<b>Relation old publications- user</b>	$(0.5-\beta-\mu)/2$	$(0.4-\beta-\mu)/2$	$(0.35-\beta-\mu)/2$
<b>Relation old shared publications - user</b>	$(0.5-\beta-\mu)/2$	$(0.4-\beta-\mu)/2$	$(0.35-\beta-\mu)/2$
<b>Quality</b>	0,05	0,1	0,14
<b>Type</b>	0,1	0,05	0,01
<b>Relation topic- interests</b>	$\alpha (=0,35 \text{ at } t=0)$	$\alpha (=0,45 \text{ at } t=0)$	$\alpha (=0,5 \text{ at } t=0)$
<b>Relation topic -old publications</b>	$0,35-\alpha$	$0,45-\alpha$	$0,5-\alpha$
<b>Penalty</b>	0.15	0.15	0.15

La puntuació per un usuari de 22 anys es calcularà de la següent forma:

$$\begin{aligned}
 score = & \beta * degree + \mu * weight + \frac{(0.5 - \beta - \mu)}{2} * relation\_oldpub\_user \\
 & + \frac{(0.5 - \beta - \mu)}{2} * relation\_oldshared\_user + 0.05 * quality + 0.1 * type \\
 & + \alpha * relation\_topic\_interests + (0.35 - \alpha) * relation\_topic\_oldpub \\
 & - Penalty
 \end{aligned}$$

- b) L'usuari C ha tingut l'oportunitat d'interaccionar amb distància 1 i 2

Es tindrà en compte majoritàriament la puntuació de distància 1 i una mica el resultat de distància 2 i la relació amb l'usuari que li ha compartit la publicació. No es fa novament separació per edat:

Rate	Age
	<b>18-85</b>
<b>Score depth 1</b>	0.6
<b>Score depth 2</b>	0.25
<b>Weight_UserB_UserC</b>	0.05
<b>Relation old shared publications - user</b>	0.1
<b>Penalty</b>	0.15

Per a qualsevol usuari es calcularà la puntuació de la següent manera:

$$score = 0.6 * score_{d1} + 0.25 * score_{d2} + 0.05 * weight + 0.1 * relation_{oldshare_{user}} - \text{Penalty}$$

- c) L'usuari C ha tingut l'oportunitat d'interaccionar amb distància 1 o 2  
 Es tindrà en compte majoritàriament la puntuació a distància i, una mica la relació amb l'usuari que li ha compartit la publicació:

Rate	Age
	<b>18-85</b>
<b>Score depth i</b>	0.7
<b>Weight_UserB_UserC</b>	0.1
<b>Relation old shared publications - user</b>	0.2
<b>Penalty</b>	0.15

Per a qualsevol usuari es calcularà la puntuació de la següent manera:

$$score = 0.7 * score_{di} + 0.1 * weight + 0.2 * relation_{oldshare_{user}} - \text{Penalty}$$

Per a tots els casos:

La probabilitat que l'usuari A faci like a la publicació 1 és igual la score obtinguda. La probabilitat de que també faci un comentari és score-0,22. La probabilitat de que també

faci un comentari és score-0,25. Tot i això hi ha un score mínim necessari per tal de que l'usuari interactuï. Per defecte és 0,3.

La probabilitat que l'usuari no interactuï és 1-score.

A l' *Annex D: Pseudo-codi de la solució dinàmica* es troba el pseudo-codi de la solució dinàmica.

A l' *Annex E: Propagació de la informació amb la versió dinàmica* trobem un exemple complert de la propagació de la informació, fent tots els càlculs d'scores descrits anteriorment.

## Annex D: Pseudo-codi de la solució dinàmica

A continuació es mostrà el pseudo-codi de la solució dinàmica.

**Algorithm 1:** genPublicationDV

```
define T, k;
for t in range(0, T) do
    n := number random between (0, k);
    users.time.t := select k random users;
    time.rate := t/T; for user in users.time.t do
        k Publications are generated;
        for publication in publications.time.t do
            user.pub = user who did the publication;
            for user.d1 in user.pub.friends do
                score1 := scoreFunction_depth1(user.d1, publication, time.rate);
                interaction.d1 := interactionFunction(score1);
                if interaction.d1 == 3 then
                    user.d1 do: like, comment and share;
                    for user.d2 in user.d1.friends do
                        score2 := scoreFunction_depth2(user.d2, publication, time.rate);
                        interaction.d2 := interactionFunction(score2);
                        if interaction.d2 == 3 then
                            user.d2 do: like, comment and share;
                            for user.d3 in user.d2.friends do
                                score3 := scoreFunction_depth2(user.d3, publication, time.rate);
                                interaction.d3 := interactionFunction(score3);
                                if interaction.d3 == 3 or interaction.d3 == 2 then
                                    user.d3 do: like and comment; #no share possibility with depth 3
                                else if interaction.d3 == 1 then
                                    user.d3 do: like
                                else
                                    user.d3 do: nothing
                        else if interaction.d2 == 2 then
                            user.d2 do: like and comment
                        else if interaction.d2 == 1 then
                            user.d2 do: like
                        else
                            user.d2 do: nothing
                else if interaction.d1 == 2 then
                    user.d1 do: like and comment
                else if interaction.d1 == 1 then
                    user.d1 do: like
                else
                    user.d1 do: nothing
```

---

**Algorithm 2:** scoreFunction\_depth1

---

```
input: user, publication, time_rate;
output: score (float between 0,1);
beta := 0.15*time_rate;
tagged_rate = 0.05;
if user.age ≥ 18 and user.age ≤ 25 then
    alpha = 0.35*time_rate; mu = 0.35*time_rate;
    degree_rate := beta; weight_rate := mu;
    relation_olddub_user := 0.5 - beta - mu; quality_rate := 0.05;
    type_rate := 0.1; relation_rate := alpha;
    relation_topic_olddub := 0.35-alpha;
else if user.age ≥ 26 and user.age ≤ 45 then
    alpha = 0.45*time_rate; mu = 0.25*time_rate;
    degree_rate := beta; weight_rate := mu;
    relation_olddub_user := 0.4 - beta - mu; quality_rate := 0.1;
    type_rate := 0.05; relation_rate := alpha;
    relation_topic_olddub := 0.45-alpha;
else
    alpha = 0.5*time_rate; mu = 0.2*time_rate
    ;
    degree_rate := beta; weight_rate := mu;
    relation_olddub_user_rate := 0.35 - beta - mu; quality_rate := 0.14;
    type_rate := 0.01; relation_rate := alpha;
    relation_topic_olddub_rate := 0.5-alpha;
if publication.type == "Image" then
    type := 1;
else if publication.type == "Video" then
    type := 0.5;
else
    type := 0.0;
if user.like1 or user.like2 or user.like3 == publication.topic then
    relation := 1;
else
    relation := 0;
if user in publication.tag then
    tagged = 1
else
    tagged = 0
relation_olddub_user := num_interactions/publication.user.total_publications;
relation_topic_olddub := num_interactions_topic/total_possible_interactions;
score := degree_rate*degree + weight_rate*weight + relation_olddub_user_rate
    *relation_olddub_user + quality_rate * quality + type_rate * type + relation_rate * relation +
    relation_topic_olddub_rate * relation_topic_olddub + tagged_rate * tagged;
return score
```

---

---

**Algorithm 3:** scoreFunction\_depth2

---

```
input: user, publication, time_rate;
output: score (float between 0,1);
if user has not the opportunity of interaction at depth 1 then
    score := scoreFunction_depth2_no(user_d2, publication, time_rate)
else
    score := scoreFunction_depth2_yes(user_d2, publication)
return score
```

---

---

**Algorithm 4:** scoreFunction\_depth2\_no

---

```
input: user, publication, time_rate;
output: score (float between 0,1);
beta := 0.15*time_rate;
penalty := 0.1;
if user.age ≥ 18 and user.age ≤ 25 then
    alpha = 0.35*time_rate; mu = 0.35*time_rate;
    degree_rate := beta; weight_rate := mu;
    relation_olddub_user := relation_olddshared_user := (0.5 - beta - mu)/2; quality_rate :=
        0.05; type_rate := 0.1;
    relation_rate := alpha; relation_topic_olddub := 0.35-alpha;
end
else if user.age ≥ 26 and user.age ≤ 45 then
    alpha = 0.45*time_rate; mu = 0.25*time_rate;
    degree_rate := beta; weight_rate := mu;
    relation_olddub_user := relation_olddshared_user := (0.4 - beta - mu)/2; quality_rate := 0.1;
    type_rate := 0.05;
    relation_rate := alpha; relation_topic_olddub := 0.45-alpha;
end
else
    alpha = 0.5*time_rate; mu = 0.2*time_rate
    ; degree_rate := beta; weight_rate := mu;
    relation_olddub_user := relation_olddshared_user := (0.35 - beta - mu)/2; quality_rate :=
        0.14; type_rate := 0.01; relation_rate := alpha; relation_topic_olddub_rate := 0.5 - alpha;
end
if publication.type == "Image" then
    | type := 1;
end
else if publication.type == "Video" then
    | type := 0.5;
end
else
    | type := 0;
end
if user.like1 or user.like2 or user.like3 == publication.topic then
    | relation := 1;
end
else
    | relation := 0;
end
relation_olddub_user := num_interactions/publication.user.total_publications;
relation_topic_olddub := num_interactions_topic/total_possible_interactions;
relation_olddshared_user := interactions_done/total_possible_interactions;
score = degree_rate*degree + weight_rate*weight + relation_olddub_user_rate *relation_olddub_user +
    relation_olddshared_user_rate * relation_olddshared_user + quality_rate * quality + type_rate *
    type + relation_rate * relation + relation_topic_olddub_rate * relation_topic_olddub - penalty;
return score
```

---

---

**Algorithm 5:** scoreFunction\_depth2\_yes

---

```
input: user, publication;
output: score (float between 0,1);
score_d1_rate := 0.7;
weight_rate = 0.1;
relation_olddshared_user := 0.2; penalty := 0.1;
relation_olddshared_user := interactions_done/total_possible_interactions;
score = score_d1_rate*score_d1 + weight_rate*weight + relation_olddshared_user_rate* relation_olddshared_user
    - penalty;
return score
```

---

---

**Algorithm 6:** scoreFunction\_depth3

---

```
input: user, publication, time_rate;
output: score (float between 0,1);
if user has the opportunity of interaction at both depth 1 and 2 then
    score := scoreFunction_depth3_1and2(user_d2, publication, time_rate)
else if user has the opportunity of interaction at depth 1 or 2 then
    score := scoreFunction_depth3_1or2(user_d2, publication, time_rate)
else
    score := scoreFunction_depth2_no(user_d2, publication, time_rate)
return score
```

---

**Algorithm 7:** scoreFunction\_depth3\_1and2

---

```
input: user, publication;
output: score (float between 0,1);
score_d1_rate := 0.6;
score_d2_rate := 0.25;
weight_rate = 0.05;
relation_oldshared_user := 0.1; penalty := 0.15;
relation_oldshared_user := interactionsd1/totalpossibleinteractions;
score = score_d1_rate*score_d1 + score_d2_rate*score_d2 + weight_rate*weight +
    relation_oldshared_user*relation_oldshared_user - penalty;
return score
```

---

**Algorithm 8:** scoreFunction\_depth3\_1or2

---

```
input: user, publication;
output: score (float between 0,1);
score_di_rate := 0.7;
weight_rate = 0.1;
relation_oldshared_user := 0.2; penalty := 0.15;
relation_oldshared_user := interactionsd1/totalpossibleinteractions;
score = score_di_rate*score_di + weight_rate*weight + relation_oldshared_user*relation_oldshared_user -
    penalty;
return score
```

---

---

**Algorithm 9:** scoreFunction\_depth3\_no

---

```
input: user, publication, time_rate;
output: score (float between 0,1);
beta := 0.15*time_rate;
penalty := 0.15;
if user.age  $\geq$  18 and user.age  $\leq$  25 then
    alpha = 0.35*time_rate; mu = 0.35*time_rate;
    degree_rate := beta; weight_rate := mu;
    relation_olddub_user := relation.oldshared.user := (0.5 - beta - mu)/2; quality_rate :=
        0.05; type_rate := 0.1;
    relation_rate := alpha; relation_topic_olddub := 0.35-alpha;
else if user.age  $\geq$  26 and user.age  $\leq$  45 then
    alpha = 0.45*time_rate; mu = 0.25*time_rate;
    degree_rate := beta; weight_rate := mu;
    relation_olddub_user := relation.oldshared.user := (0.4 - beta - mu)/2; quality_rate := 0.1;
    type_rate := 0.05;
    relation_rate := alpha; relation_topic_olddub := 0.45-alpha;
else
    alpha = 0.5*time_rate; mu = 0.2*time_rate
    ; degree_rate := beta; weight_rate := mu;
    relation_olddub_user := relation.oldshared.user := (0.35 - beta - mu)/2; quality_rate :=
        0.14; type_rate := 0.01; relation_rate := alpha; relation_topic_olddub := 0.5 - alpha;
if publication.type == "Image" then
    type := 1;
else if publication.type == "Video" then
    type := 0.5;
else
    type := 0.0;
if user.like1 or user.like2 or user.like3 == publication.topic then
    relation := 1;
else
    relation := 0;
relation_olddub_user := num_interactions/publication.user.total_publications;
relation_topic_olddub := num_interactions/topic/total_possible_interactions;
relation_oldshared_user := interactions_done/total_possible_interactions;
score = degree_rate*degree + weight_rate*weight + relation_olddub_user_rate *relation_olddub_user +
    relation_oldshared_user_rate * relation.oldshared.user + quality_rate * quality + type_rate *
    type + relation_rate * relation + relation_topic_olddub_rate * relation_topic_olddub - penalty;
return score
```

---

---

**Algorithm 10:** interactionFunction

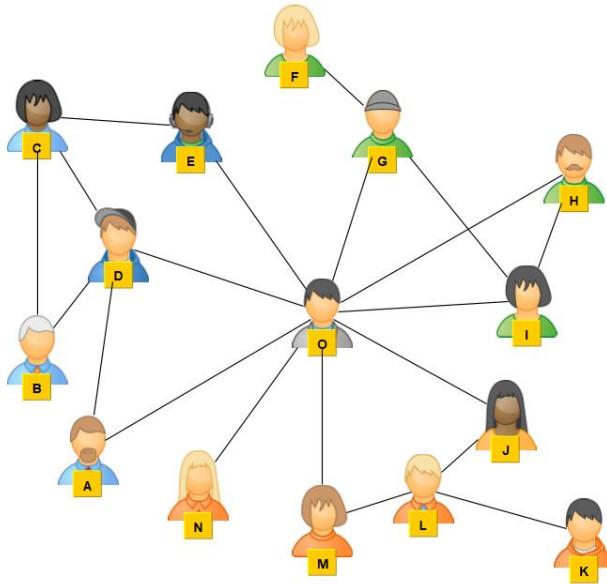
---

```
input: score;
output: int that can be 0, 1 or 2;
rand_num1 := random_number between (0,1);
if score  $\geq$  rand_num1 and score  $\geq$  0,3 then
    rand_num2 := random_number between (0,1);
    if score - 0.25  $\geq$  rand_num2 then
        return 3;
    else if score - 0.22  $\geq$  rand_num2 then
        return 2;
    else
        return 1;
    else
        return 0;
```

---

## Annex E: Propagació de la informació amb la versió dinàmica

A continuació es mostra un exemple complert de com es produeix la propagació en el programa. Consta d'un graf de 15 nodes (usuaris) que estan connectats entre ells de la següent manera:



D. Graf de l'exemple

L'usuari A farà la següent publicació:

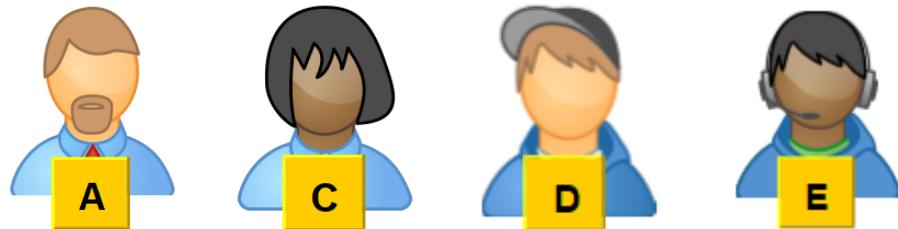
User	A
ID_publication	1
Type	Image
Publication_description	Kid dancing on the Great Wall of China
Topic	Travels
Quality	0,6
Users_tagged	User_O

L'exemple tracta la solució amb historial i es produeix en la següent situació:

- Temps màxim marcat: 30 unitats
- Moment on es realitza la publicació: 13 unitats
- Durant aquest temps, els usuaris han anat interactuant amb les publicacions i han creat els atributs *Relation oldPublications-user* i *relation topic-oldPublications*.

Els usuaris més importants en el següent exemple tenen els atributs que es mostren a la següent taula. La fila *relation topic-oldPublications* fa referència a la relació entre l'usuari i les publicacions de Topic *Travels* que s'han produït anteriorment. La fila

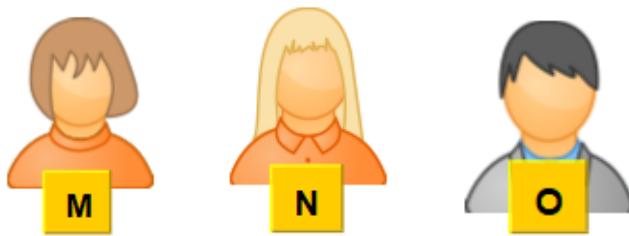
*Relation oldPublications-user* fa referència a la relació entre les anteriors publicacions de l'usuari A i l'usuari en qüestió. La fila *degree* està normalitzada tal i com s'ha explicat. Els usuaris tenen els següents atributs:



	A	C	D	E
<i>User</i>	A	C	D	E
<i>Age</i>	60	28	21	51
<i>Interest 1</i>	Music artist	Music artist	Sports	Travels
<i>Interest 2</i>	Food	Drink brand	Computer games	TV show
<i>Profession</i>	Manager	Service	Student	Professional
<i>Degree</i>	0,86	0,71	0,57	0,86
<i>Relation topic-oldPublications</i>	0,8	0,76	0,3	0,83
<i>Relation</i>	-	0,21	0,14	0,34
<i>oldPublications-user</i>				



	G	H	I	J
<i>User</i>	G	H	I	J
<i>Age</i>	32	28	46	21
<i>Interest 1</i>	News and present	Animals	Sports	Travels
<i>Interest 2</i>	TV show	Food	Computer games	News and present
<i>Profession</i>	Sales and office	Construction	Transportation	Manager
<i>Degree</i>	0,71	0,86	0,71	0,86
<i>Relation topic-oldPublications</i>	0,4	0,79	0,34	0,17
<i>Relation</i>	0,11	0,2	0,15	0,1
<i>oldPublications-user</i>				



User	M	N	O
Age	26	61	35
Interest 1	Travels	Food	Travels
Interest 2	Drink brand	Computer games	Food
Profession	Maintance	Service	Manager
Degree	0,86	1	0
Relation topic-oldPublications	0,67	0,72	0,86
Relation oldPublications-user	0,51	0,42	0,78

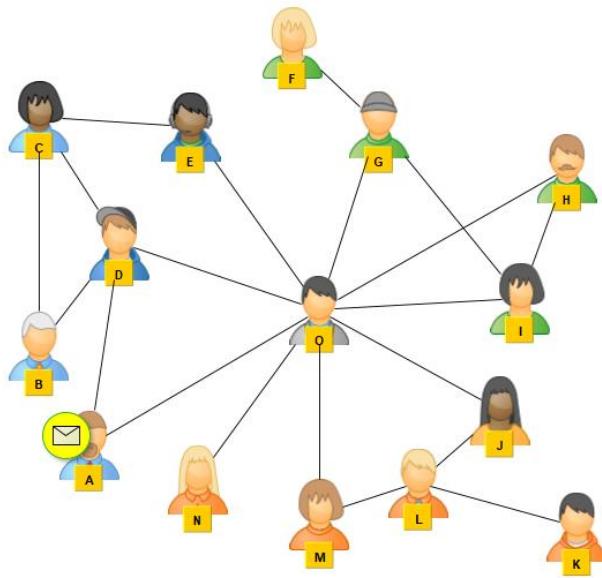
Una observació interessant és la següent: clarament l'usuari O té moltes més connexions que la resta, considerant-se així un influencer.

A continuació es mostra la taula de weights entre els usuaris descrits anteriorment:

WEIGHTS	A	C	D	E	G	H	I	J	M	N	O
A	-	0	0,2	0	0	0	0	0	0	0	0,75
C	0	-	0,5	0,67	0	0	0	0	0	0	0
D	0,2	0,5	-	0	0	0	0	0	0	0	0,3
E	0	0,67	0	-	0	0	0	0	0	0	0,82
G	0	0	0	0	-	0	0,6	0	0	0	0,31
H	0	0	0	0	0	-	0,21	0	0	0	0,77
I	0	0	0	0	0,6	0,21	-	0	0	0	0,15
J	0	0	0	0	0	0	0	-	0	0	0,23
M	0	0	0	0	0	0	0	0	-	0	0,65
N	0	0	0	0	0	0	0	0	0	-	0,81
O	0,75	0	0,3	0,82	0,31	0,77	0,15	0,23	0,65	0,81	-

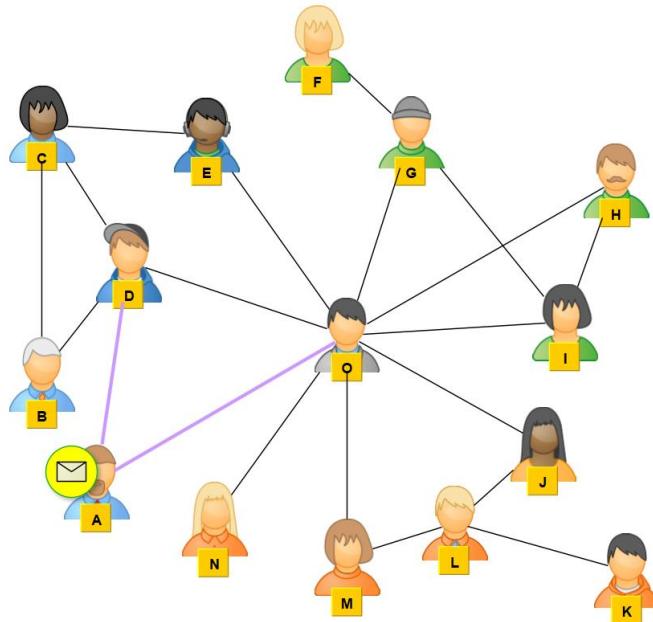
Podem començar doncs amb la propagació de la informació:

1. Usuari A crea la publicació 1 descrita anteriorment:



D. Creació publicació 1

2. La publicació arriba als usuaris D i O, que són els únics que estan connectats amb distància 1 amb l'usuari que crea la publicació:



D. Propagació distància 1

3. El programa utilitza els algoritmes descrits a l' Annex D: *Pseudo-codi de la solució dinàmica* per calcular l'score entre cadascun d'aquests usuaris i la publicació 1. Les formules que expliquen com es calcula l'score segons la distància a la que es trobin els usuaris es troben a l'apartat *Solució dinàmica*. A continuació es detallarà com es produeix el càlcul per a distància 1:

	Age		
	18-25	26-45	46-85
<b>Degree</b>	$\beta (=0,15 \text{ at } t=0)$	$\beta (=0,15 \text{ at } t=0)$	$\beta (=0,15 \text{ at } t=0)$
<b>Weight</b>	$\mu (=0,35 \text{ at } t=0)$	$\mu (=0,25 \text{ at } t=0)$	$\mu (=0,2 \text{ at } t=0)$
<b>Relation old publications- user</b>	$0.5 - \beta - \mu$	$0.4 - \beta - \mu$	$0.35 - \beta - \mu$
<b>Quality</b>	0,05	0,1	0,14
<b>Type</b>	0,1	0,05	0,01
<b>Relation topic- Interest</b>	$\alpha (=0,35 \text{ at } t=0)$	$\alpha (=0,45 \text{ at } t=0)$	$\alpha (=0,5 \text{ at } t=0)$
<b>Relation topic -old publications</b>	$0,35 - \alpha$	$0,45 - \alpha$	$0,5 - \alpha$
<b>Tagged (extra)</b>	0,05	0,05	0,05

Per a l'usuari D, qui té 21 anys, l'score seria el següent:

$$\begin{aligned} \textit{score} = & \beta * \textit{degree} + \mu * \textit{weigth} + (0.5 - \beta - \mu) * \textit{relation\_oldpub\_user} + \\ & 0.05 * \textit{quality} + 0.1 * \textit{type} + \alpha * \textit{relation\_topic\_interests} + (0.35 - \alpha) * \\ & \textit{relation\_topic\_oldpub} + 0.05 * \textit>tagged \end{aligned}$$

$$\begin{aligned} \textbf{Score} = & 0,085 * 0,57 + 0,198 * 0,2 + (0,5 - 0,085 - 0,198) * 0,14 + 0,05 * 0,6 \\ & + 0,1 * 1 + 0,198 * 0 + (0,35 - 0,198) * 0,3 + 0,05 * 0 = \mathbf{0,294} \end{aligned}$$

Ara es cridaria a la funció *InteractionFunction*, qui de manera aleatòria tenint en compte aquest score decidiria si l'usuari interacciona o no:

---

**Algorithm 10:** interactionFunction

---

```

input: score;
output: int that can be 0, 1 or 2;
rand_num1 := random_number between (0,1);
if score ≥ rand_num1 and score ≥ 0,3 then
    rand_num2 := random_number between (0,1);
    if score - 0,25 ≥ rand_num2 then
        ↵ return 3;
    else if score - 0,22 ≥ rand_num2 then
        ↵ return 2;
    else
        ↵ return 1;
else
    ↵ return 0;

```

---

Recordem que cada return significa el següent:

- Return 0: no interaction
- Return 1: Like
- Return 2: Like and Comment

- Return 3: Like, Comment and Share

Utilitzarem un script de Python per generar el número aleatori. Ens surt  $rand\_num1 = 0,43$  de manera que retorna 0 i per tant no interactua.

Els valors que no s'han indicat a les taules definides anteriorment s'han calculat de la següent manera:

- Com ens trobem a la unitat temporal 13 sobre 30, els valors de  $\beta$ ,  $\mu$  i  $\alpha$  són els següents:

- $B = 0,15 - \frac{13}{30} * 0,15 = 0,085$
- $\mu = 0,35 - \frac{13}{30} * 0,35 = 0,198$
- $\alpha = 0,35 - \frac{13}{30} * 0,35 = 0,198$

- Type = 1 ja que és tracta d'una publicació de tipus Image.
- Relation = 0 ja que cap dels interests de l'usuari D és Food.
- Tagged = 0 ja que no apareix a la imatge.

Per a l'usuari O, qui té 35 anys, calcularem l'score de la següent manera:

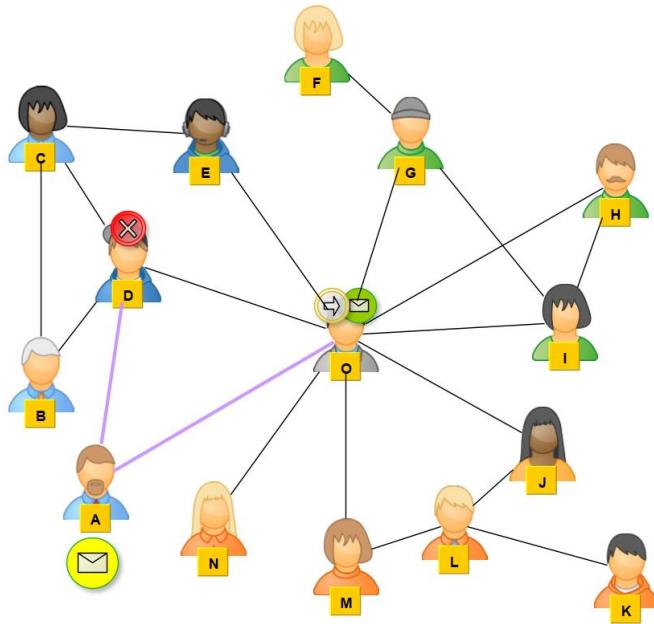
$$\begin{aligned} score = & \beta * degree + \mu * weight + (0.4 - \beta - \mu) * relation\_oldpub\_user + 0.1 \\ & * quality + 0.05 * type + \alpha * relation\_topic\_interests \\ & + (0.45 - \alpha) * relation\_topic\_oldpub + 0.05 * tagged \end{aligned}$$

On

- $B = 0,15 - \frac{13}{30} * 0,15 = 0,085$
- $\mu = 0,25 - \frac{13}{30} * 0,25 = 0,141$
- $\alpha = 0,45 - \frac{13}{30} * 0,45 = 0,255$

$$\begin{aligned} score = & 0,085 * 0 + 0,141 * 0,75 + (0.4 - 0,085 - 0,141) * 0,78 + 0.1 * 0,6 + \\ & 0.05 * 1 + 0,255 * 1 + (0.45 - 0,255) * 0,86 + 0.05 * 1 = \mathbf{0,824} \end{aligned}$$

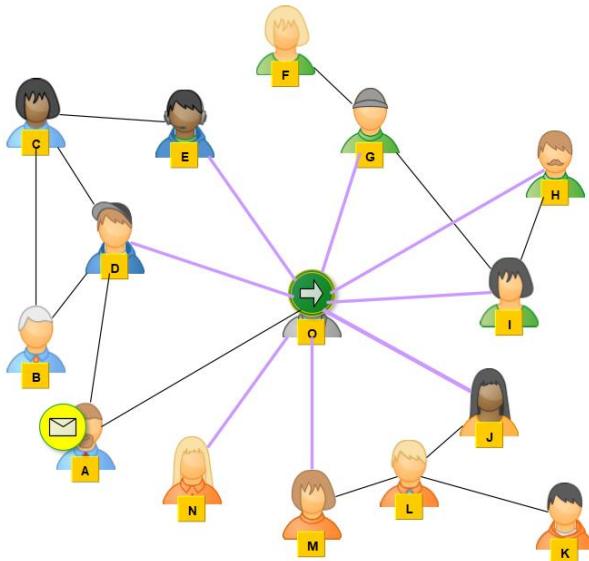
Ens surt  $rand\_num=0,62$  i  $rand\_num2=0,36$ . De manera que l'usuari dona like, comenta i fa share.



D. Decisió interacció

En les pròximes parts s'ha utilitzat el mateix procediment que en aquesta amb l'algorisme corresponent.

4. L'usuari O comparteix (*fa share*) la publicació. Aquesta arriba per tant als usuaris: D, E, G, H, I, J, M, N. Podem veure com al tractar-se d'un influencer fa arribar la publicació a molts usuaris que no estan connectats amb l'usuari A a distància 1.



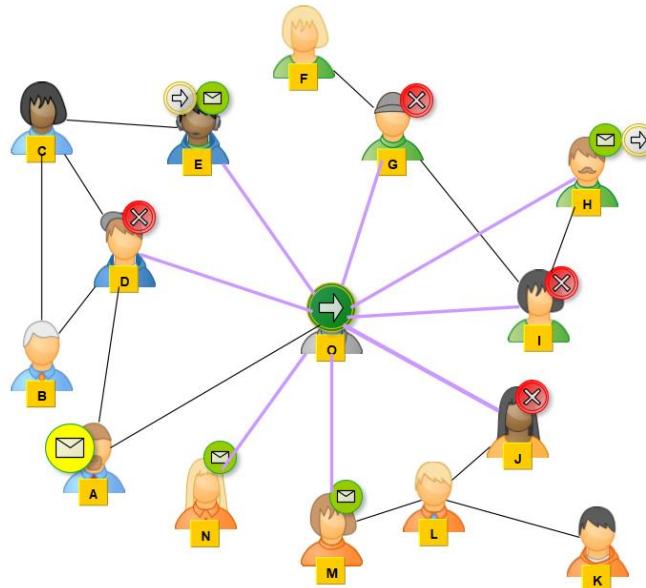
D. Propagació informació usuari O

5. De la mateixa manera que abans, per a cadascun dels usuaris s'utilitza la formula per calcular l'score (ara les formules amb distància 2) i es crida a la funció *InteractionFunction* per decidir si s'interactua i de quina manera amb la publicació.

Es té en compte que l'usuari D ha tingut prèviament l'oportunitat d'interactuar, mentre que els demes usuaris, encara no.

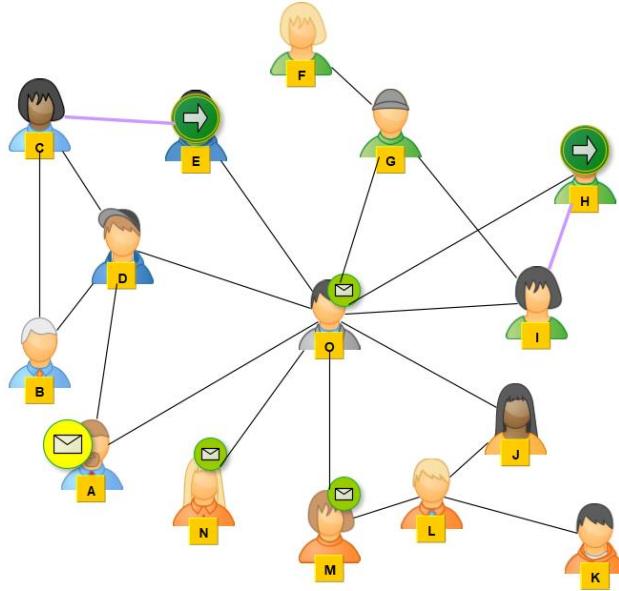
Després de calcular els scores i cridar a *InteractionFunction* obtenim:

- Usuari D: no interacciona.
- Usuari E: like, comment and share.
- Usuari G: no interacciona.
- Usuari H: like, comment and share.
- Usuari I: no interacciona.
- Usuari J: no interacciona.
- Usuari M: Like and comment.
- Usuari N: Like.



#### D. Decisió d'interacció

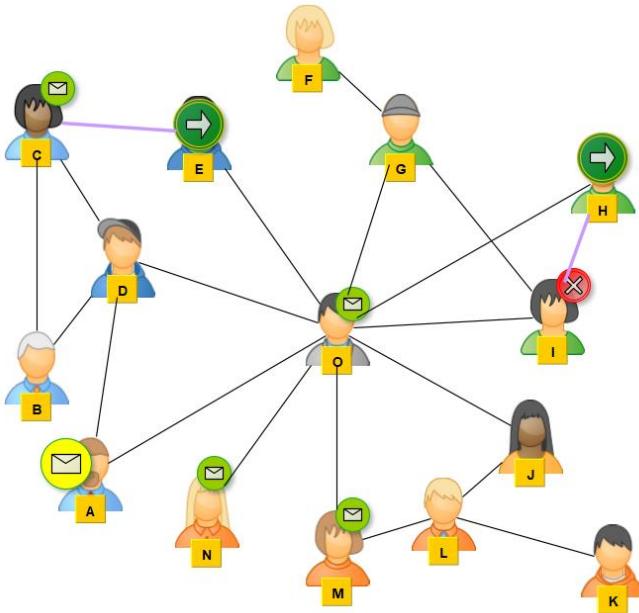
6. Els usuaris E i H, al haver compartit la publicació, fan que aquesta arribi als usuaris C i I, respectivament.



D. Propagació informació usuaris E i H

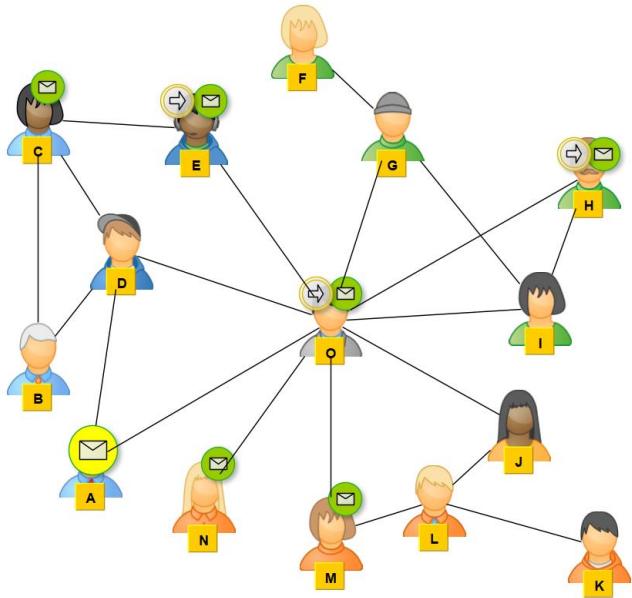
7. Es calcula l'score per als usuaris C i I tenint en compte que cap dels dos havia tingut oportunitat anteriorment d'haver interaccionat amb la publicació. Després de calculat l'score i cridat a la *InteractionFunction*, obtenim el següent:

  - Usuari C: Like.
  - Usuari I: no interacciona.



D. Decisió interacció

8. Finalment, després de la propagació de la informació, s'han obtingut les següents interaccions:



D. Resultat final

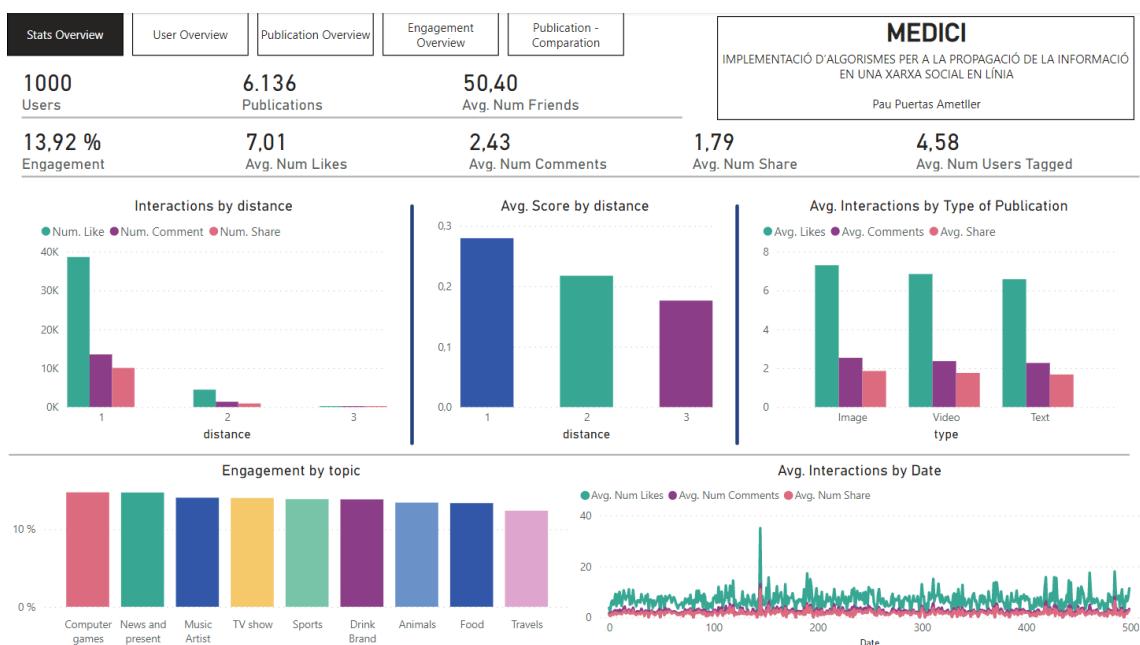
## Annex F: Taulell Power BI

El *dashboard* fet amb Power BI ens permet accedir a les dades de manera interactiva. D'aquesta forma podrem analitzar i treure les nostres pròpies conclusions de la simulació.

Consta de 5 pantalles diferents que són explicades a continuació:

### a) Stats Overview

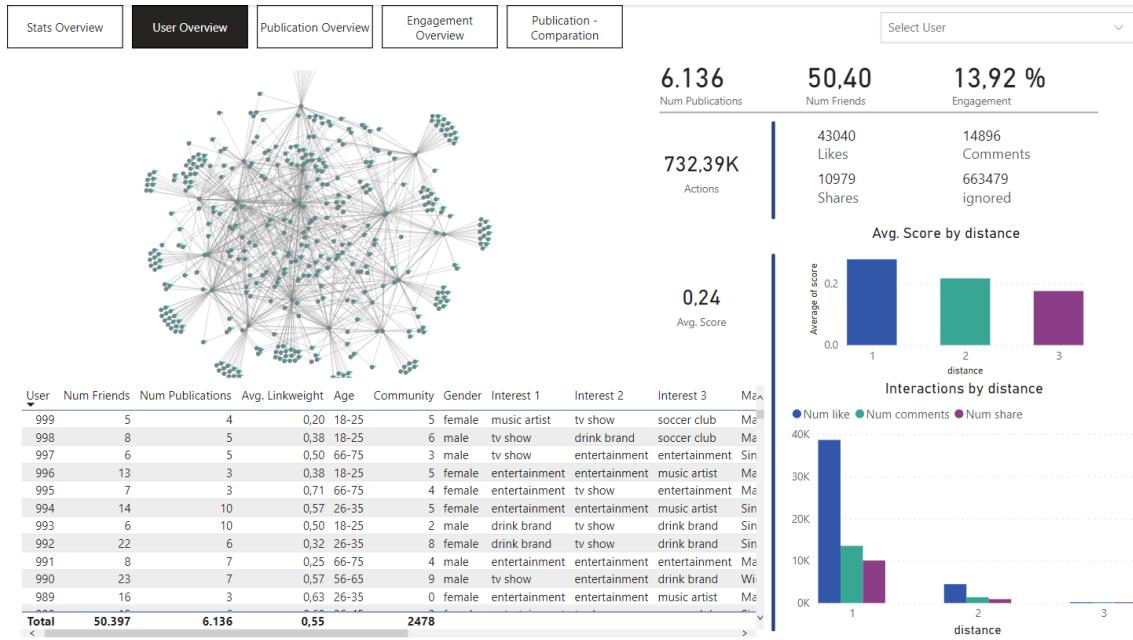
Es pot consultar un resum de les estadístiques generades. Es mostren els KPIs més importants i gràfics d'altres pantalles que s'expliquen a continuació. Ens permet fer-nos una idea ràpida de com ha anat l'execució.



E. Power BI - Stats Overview

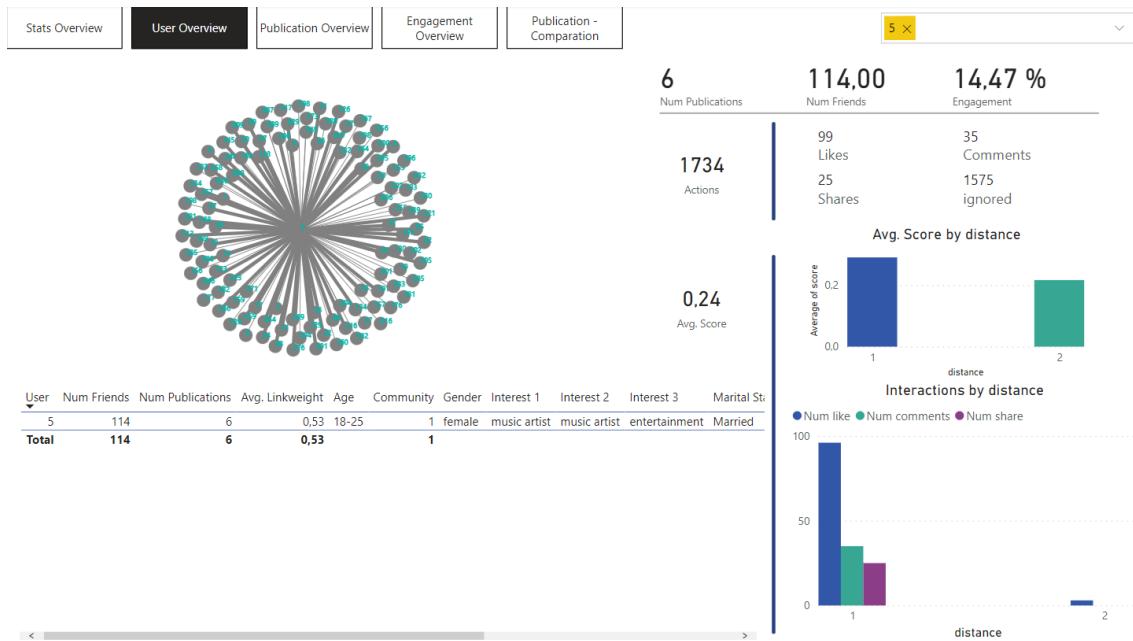
### b) User Overview

Es pot consultar el graf que s'ha generat en la simulació. Es mostra un resum de les publicacions totals creades, el número d'amics mitjà i la mitjana d'Engagement Rate. També, es mostra el total d'accions fets i de quin tipus han estat: Like, comentar, compartir o ignorar. Una mica més avall es mostra la mitjana d'score que ha generat el programa amb els algorismes i funcions plantejats anteriorment, segmentat entre les diferents distàncies. Per últim es mostra una taula amb tota la informació rellevant dels usuaris de la xarxa social, com el nombre d'amics, la comunitat o l'edat entre altres.



E. Power BI - User Overview

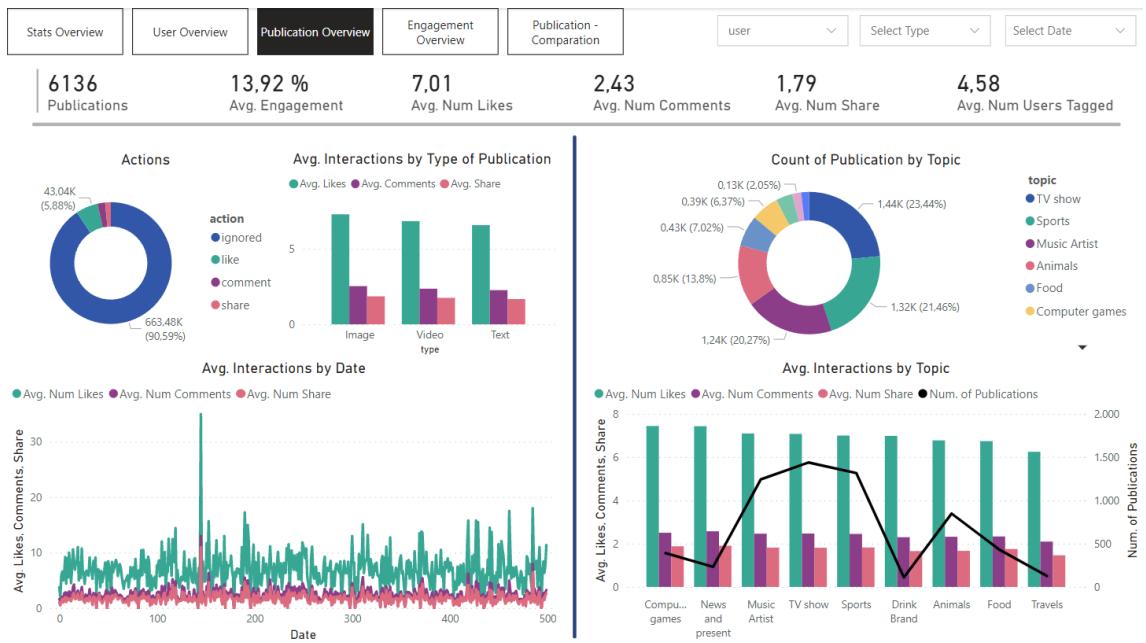
Es pot filtrar per usuari (o usuaris) per mostrar únicament la informació i valors resum d'aquest:



E. Power BI - User Overview amb filtre

### c) Publication Overview:

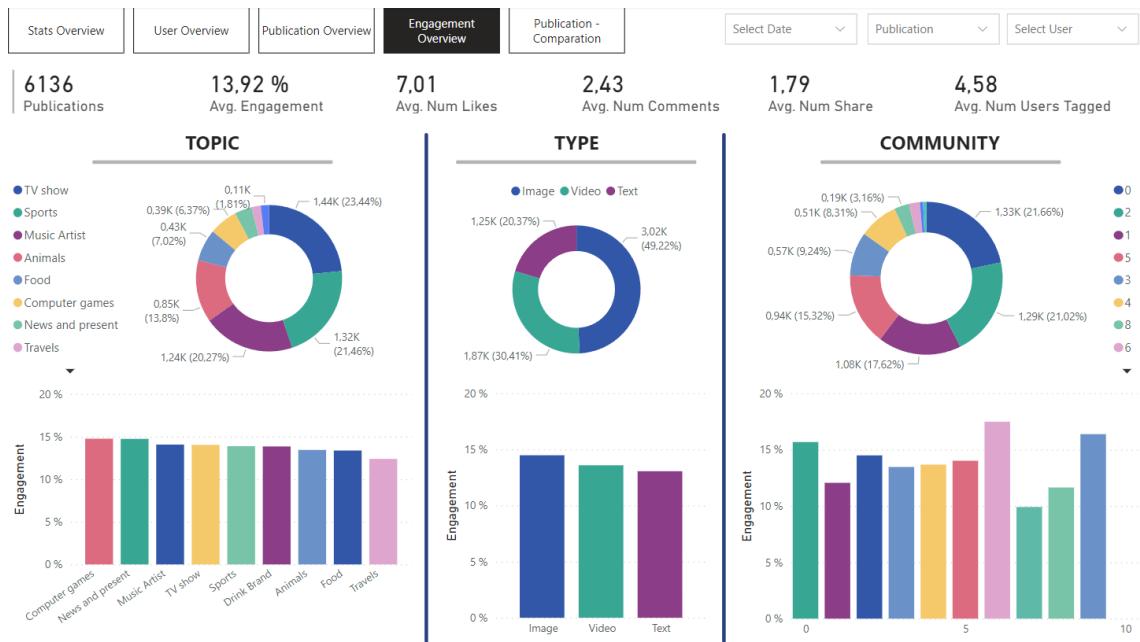
Permet consultar el nombre de publicacions i la mitjana d'Engagement Rate, de les diferents interaccions per publicació i dels usuaris etiquetats. Es pot veure un gràfic de les diferents accions produïdes segmentant per tipus i topic de publicació. També les interaccions produïdes per data i el nombre de publicacions de cadascun dels temes. Es pot filtrar per usuari, per tipus i per data per obtenir informació més concreta.



E. Power BI - Publications Overview

#### d) Engagement Overview:

Permet consultar l'Engagement Rate diferenciat per Topic, Type i Community. Podem filtrar per data, publicació i usuari.

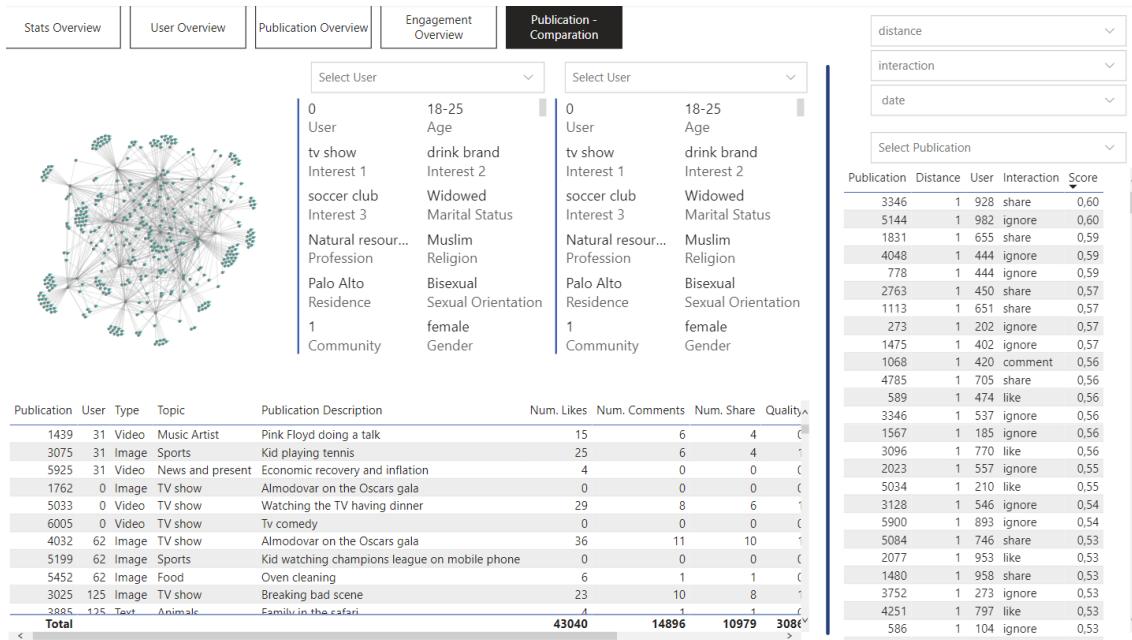


E. Power BI - Engagement Overview

#### e) Publication - Comparition

Pantalla molt útil per a poder comparar usuaris i entendre perquè dos usuaris estan fortament connectats o han interaccionat. Podem filtrar dos usuaris diferents i veure les seves característiques i atributs. Sens mostra una taula amb la informació de cada

publicació i una altre on sens mostren totes les interaccions produïdes. D'aquesta forma podem trobar usuaris que han interactuat en una publicació, amb quin score i fer la comparativa entre ells.



E. Power BI - Publications comparition

## Annex G: Fitxers d'entrada del programa

Ens trobem amb 3 fitxers d'entrada en el programa:

### a) Topics description

El fitxer d'entrada que permet la correcta creació de les publicacions és el següent: *topics\_descriptions.csv*. És un fitxer .csv separat per comes que té el següent format:

Music Artist	The Beatles museum
Music Artist	Elvis Presley show
Music Artist	Michael Jackson dancing
Music Artist	Led Zeppelin singing
Music Artist	Pink Floyd doing a talk
Music Artist	Eminem singing on NFL
Music Artist	Listening queen on the bed
Music Artist	AC/DC playing the guitar
Music Artist	Piano classes academy
Music Artist	Rap battle on the street
Music Artist	Viñarock tickets
Music Artist	1M of assistants in a concert
Music Artist	Boy and girl dancing
Music Artist	Family singing
Music Artist	Driving a car with music
Drink Brand	Disco dance
Drink Brand	Kid drinking Coca-Cola

F. Topics description

A la primera columna es detallen els *topics* i a la segona columna les *publications descriptions* relacionades. D'aquesta manera el programa el que fa és de manera aleatòria (segons les probabilitats marcades per l'usuari) decideix un *topic* i li assigna de manera aleatòria una descripció per aquell topic.

Es pot modificar sense problemes el fitxer d'entrada. Sempre que el format sigui el mateix per a cada fila: un topic dels que permet el programa i una descripció que estigui relacionada amb aquell tema.

Els dos següents s'utilitzen per a generar el graf:

### b) Graph Structure

Es un fitxer csv on apareixen tots els enllaços que hi ha. Consta de dues columnes i a cada fila apareixen dos nodes que estan enllaçats. Es pot crear desde la intereficie d'usuari, agafar un per defecte, o crear un personalment.

345	103
296	300
992	932
122	336
401	682
289	416
798	673
790	792
699	638
36	571

*F. Graph Structure*

### c) Community Labels

Fitxer que assigna cada node a una comunitat. Consta de dues columnes, la primera són els nodes i la segona el numero de comunitat.

0	1
1	1
2	1
3	4
4	1
5	1
6	1
7	1
8	1
9	4
10	8

*F. Community Labels*

## Annex H: Fitxers de sortida del programa

El programa genera varis fitxers de sortida que són utilitzats tant per JavaFX per les pantalles de resultats, com per PowerBi per a les diferents visualitzacions. Els 4 primers són per recollir la informació del graf i els demés per la informació corresponent a les publicacions.

### a) 1kby30k\_out:

Per veure totes les característiques per a cadascun dels usuaris de la xarxa:

user	age	gender	residence	religion	maritalstatus	profession	politicalorient	sexualorienta	numfriends	like1	like2	like3	classvalue	auth	community
999	26-35	male	Cambridge	Muslim	Married	Service	Far Left	Heterosexual	LOW	music artist	tv show	soccer club	NO	0.026178	5
998	36-45	female	Cambridge	Christian	Single	Professional	Left	Asexual	LOW	drink brand	tv show	drink brand	NO	0.041885	6
997	56-65	female	Santa Barbara	Hindu	Married	Production	Center Left	Heterosexual	LOW	tv show	entertainmen	entertainmen	NO	0.031414	3
996	18-25	female	Boston	Muslim	Married	Professional	Center Right	Heterosexual	LOW	music artist	tv show	soccer club	NO	0.068063	5
995	66-75	female	San Jose	Jewish	Married	Production	Far Left	Heterosexual	LOW	entertainmen	tv show	entertainmen	NO	0.036649	4

G. kby30k\_out

### b) 1kby30k\_out1:

Permet veure per a cada valor de cada atribut quina és la seva freqüència absoluta:

community	attribute	value	frequency
ALL	AGE	18-25	284
ALL	AGE	26-35	264
ALL	AGE	36-45	188
ALL	AGE	46-55	38
ALL	AGE	56-65	115
ALL	AGE	66-75	81
ALL	AGE	76-85	30
ALL	GENDER	Male	359
ALL	GENDER	Female	641

G. 1kby30\_out1

### c) 1kby30k\_out2:

Permet veure per a cada comunitat quin és el nombre total d'usuaris:

9	9
8	28
7	5
6	24
5	157
4	81
3	97
2	211
1	172
0	216

G. 1kby30\_out2

d) 1kby30k\_outg:

Permet veure els pesos de tots els enllaços que existeixen entre usuaris:

user	userf	linkweight
999	487	0.6141975308641976
999	658	0.7160493827160495
999	428	0.7160493827160493
999	731	0.7160493827160493

G. 1kby30\_outG

e) Info\_publications\_din (sta):

Fitxer principal de les publicacions. Se'ns mostra per a cada publicació quin usuari l'ha fet, el tipus de publicació, el topic i la descripció, la qualitat, el numero de likes, comentaris, shares, la data i el número d'usuaris etiquetats.

publication	user	type	topic	publication_c quality	num_likes	num_comme	num_share	date	users_tagged
0	713	Image	Sports	Tokio Olympic 0.195598233	8	1	0	0	1
1	147	Image	Computer gar Friends playin	0.607977439	29	8	2	0	4
2	3	Image	Music Artist	Eminem singir 0.436041308	31	4	0	0	9
3	552	Video	Animals	Dog sleeping i 0.531220358	11	2	0	0	6
4	943	Text	Food	Goulash in Hu 0.512312364	9	2	0	0	5
5	373	Video	News and pre	The rise of NF 0.228006093	6	1	0	0	0
6	778	Video	Sports	Mbappe comi 0.738474656	23	4	1	0	4

G. Info\_publications\_din

f) Numfriends\_numpub\_din:

Ens permet veure per a cada usuari quants amics té i quantes publicacions ha fet.

user	num friends	num publications
0	189	0
1	145	3
2	135	6
3	113	4
4	145	1
5	114	5

G. Numfriends\_numpub\_din

g) Score\_given\_publications\_din (sta):

Permet veure per a cada usuari, quines publicacions li han arribat, a quina distància, quin score se li ha calculat i quina interacció ha fet.

user	publication	distance	score	interaction
0	2	1	0.16031300745613083	ignore
0	1028	1	0.11981390082964752	ignore
0	2052	1	0.114698301714469	ignore
0	2054	1	0.17956855795760404	ignore
0	9	1	0.1761607833590173	ignore
0	13	1	0.17607000000000002	ignore

G. Score\_given\_publications

#### h) Tagged\_publications\_din:

Mostra per a cada publicació quins són els usuaris que han estat etiquetats.

publication	user Tagged
0	893
1	438
1	19
1	123
1	894
2	627

G. Tagged\_publications\_din

#### i) User\_publications\_din(sta):

Mostra per a cada usuari quines publicacions ha fet. Tot i que sembla informació repetida, ens permet al power BI enllaçar les taules de manera que els filtres es poden fer tant per usuari com per publicació.

user	publication
1	351
1	372
1	1108
2	88

G. User\_publications\_din

#### j) User\_interactions\_publications\_din:

Ens mostra per a cada usuari, les interaccions que ha fet amb cadascuna de les publicacions. permet al power BI enllaçar les taules amb score\_given\_publications\_din.

user	publication	action
0	226	ignored
0	306	ignored
0	434	ignored
0	613	ignored
0	738	ignored
0	814	ignored
0	841	ignored
0	874	ignored
0	982	ignored
0	1020	ignored
0	1069	ignored
0	1161	ignored
0	1178	ignored
0	1318	ignored

*G. User\_interactions\_publications\_din*

## Annex I: Explicació de les pantalles de la interfície d'usuari

Les pantalles que he generat jo des de 0 són *i) Generate Data i k) Results Publications*. En les pantalles *c) Input Files i g) Output Files Settings* he afegit noves opcions corresponents als fitxers de les publicacions. Les demés pantalles són la configuració dels paràmetres dels grafs.

### a) Portada

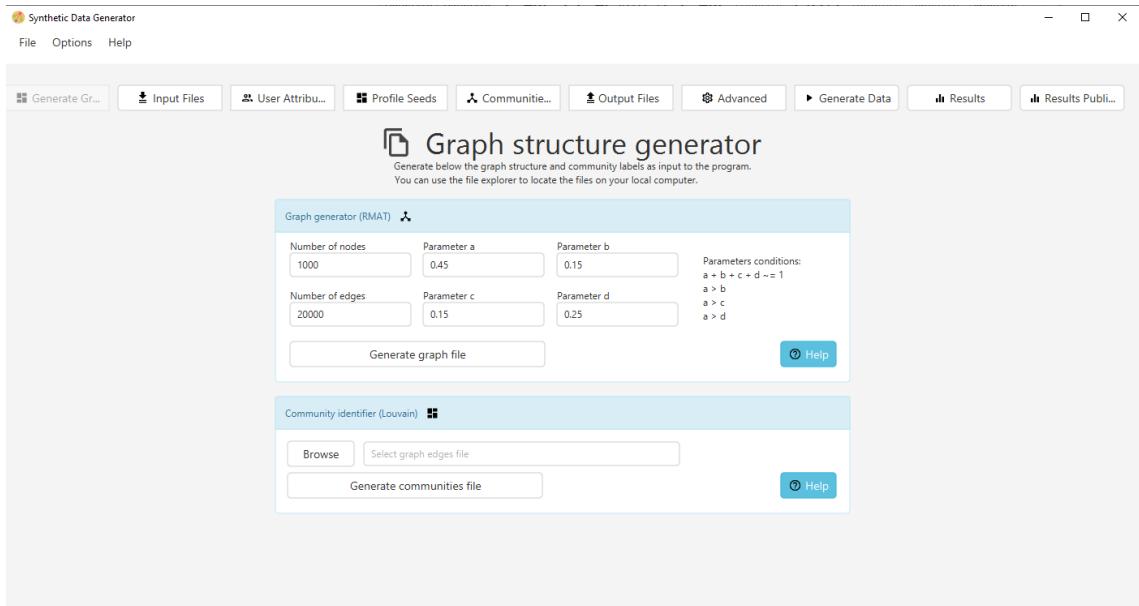
Quan executem el programa s'obre una nova finestre que ens permet començar a executar el nostre programa modificant els paràmetres:



H. Interfície d'usuari – Portada

### b) Generate Graph

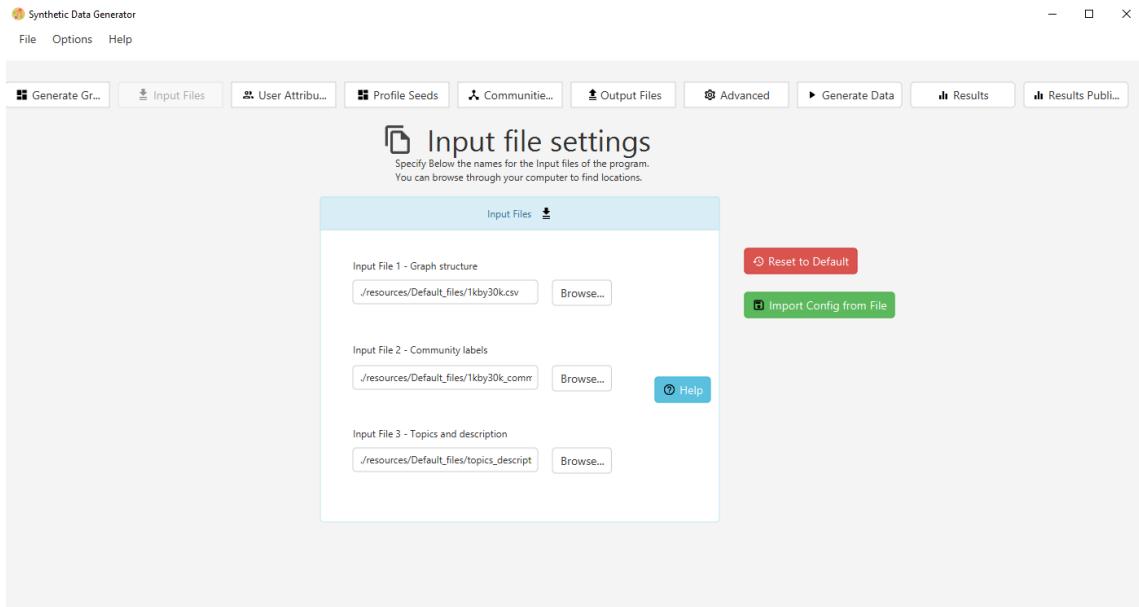
A continuació se'ns obre la pestanya Generate graph. Aquí podem seleccionar el numero de nodes i enllaços que formaran la xarxa. Hem de generar el graf i a continuació seleccionar-lo en la part de sota per a generar el fitxer de comunitats.



H. Interfície d'usuari - Generate Graph

### c) Input file Settings

Ens serveix per seleccionar els 3 fitxers d'entrada que han estat explicats abans.



H. Interfície d'usuari - Input file Settings

### d) User Attributes

Podem configurar les probabilitats de cadascun dels atributs per als usuaris.

The screenshot shows the 'User Attributes' configuration screen. At the top, there's a header with tabs: 'Generate Gr...', 'Input Files', 'User Attribu...', 'Profile Seeds' (which is selected), 'Community...', 'Advanced', 'Output Files', 'Generate Data', 'Results', and 'Results Publi...'. Below the header, the title 'User Attributes' is displayed with a subtitle: 'Here you can adjust the frequencies in which each attributes' parameters will be assigned'. The interface is divided into six sections: Age, Gender, Residence, Religion, Marital Status, and Profession. Each section contains a table with parameters and an 'Edit' button.

H. Interfície d'usuari - User Attributes

### e) Profile Seed Settings

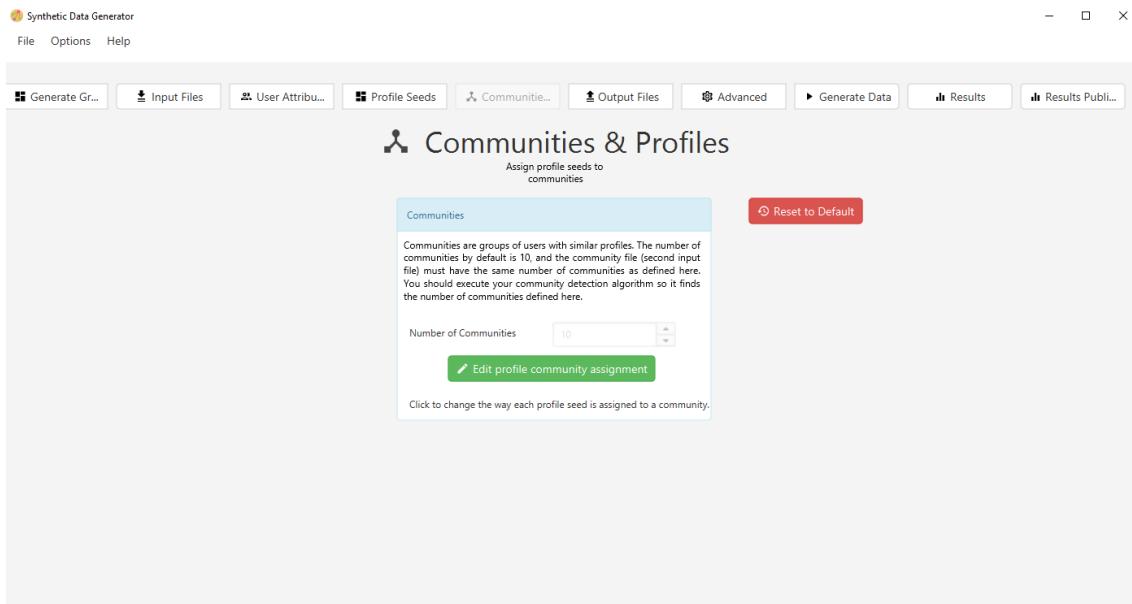
Ens serveix per configurar els perfils de les diferents llavors de cada comunicat.

The screenshot shows the 'Profile Seed Settings' configuration screen. At the top, there's a header with tabs: 'Generate Gr...', 'Input Files', 'User Attribu...', 'Profile Seeds' (selected), 'Community...', 'Output Files', 'Advanced', 'Generate Data', 'Results', and 'Results Publi...'. Below the header, the title 'Profile Seed Settings' is displayed with a subtitle: 'Define profiles: assign attribute-values to profile seeds.' There are five profile settings labeled Profile 0 through Profile 5, each containing dropdown menus for various attributes like age, gender, religion, and profession. A red 'Reset to Default' button is located at the top right of the profile area.

H. Interfície d'usuari - Profile Seed Settings

### f) Communities & Profiles

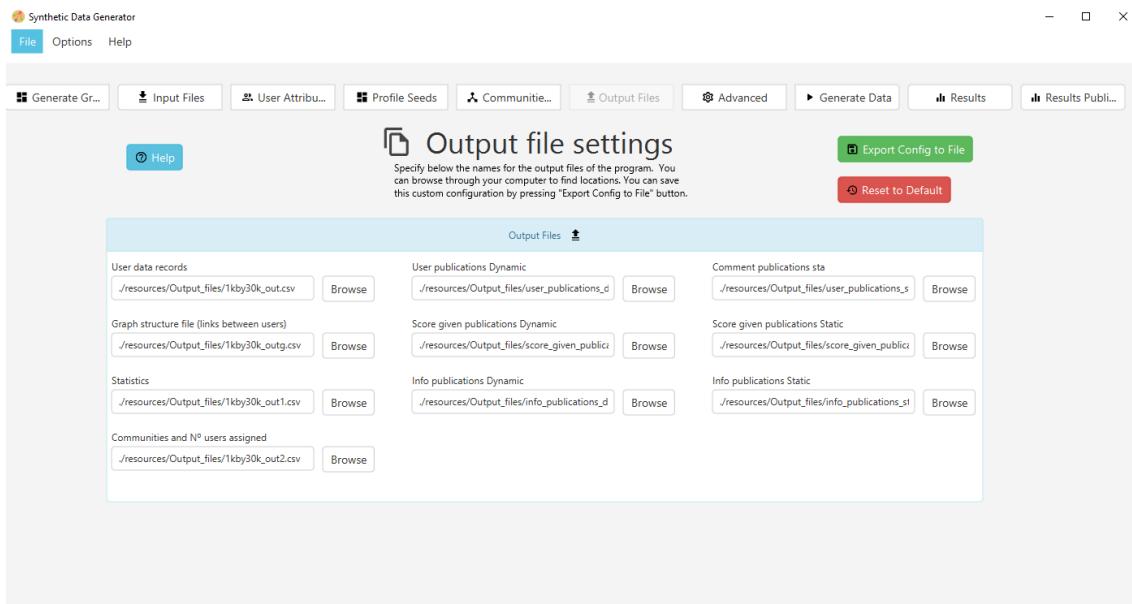
Ens permet assignar cada llavor a cada comunitat.



*H. Interfície d'usuari - Communities & Profiles*

### g) Output file Settings

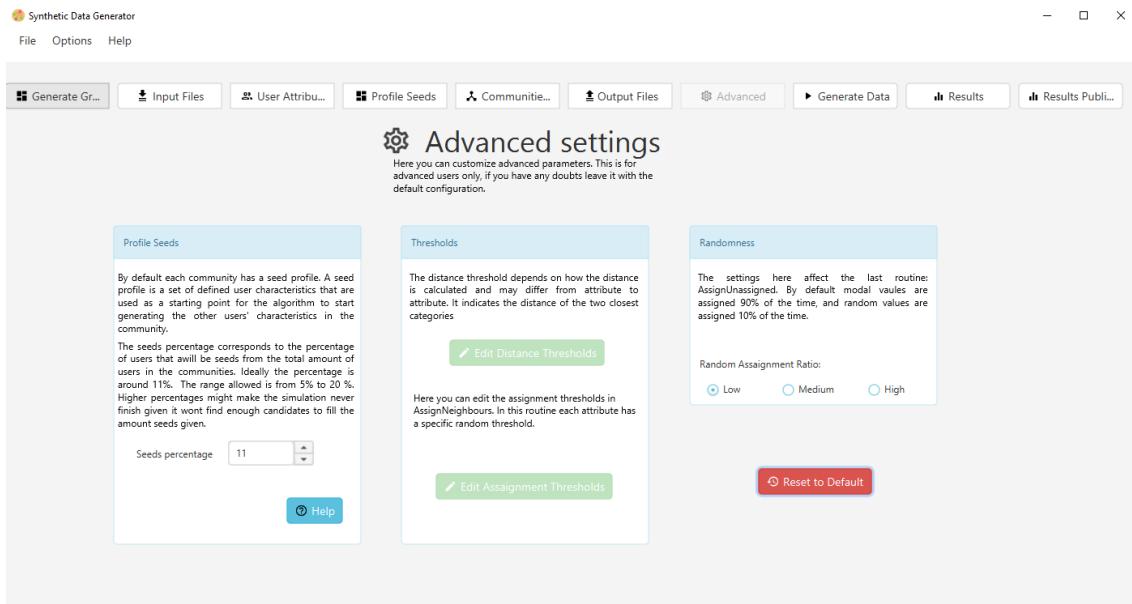
Ens permet seleccionar la ruta i el nom on volem que es generin els fitxers de sortida del programa. Aquests són utilitzats al PowerBI i ens permeten veure tota la informació generada.



*H. Interfície d'usuari - Output file settings*

## h) Advanced Settings

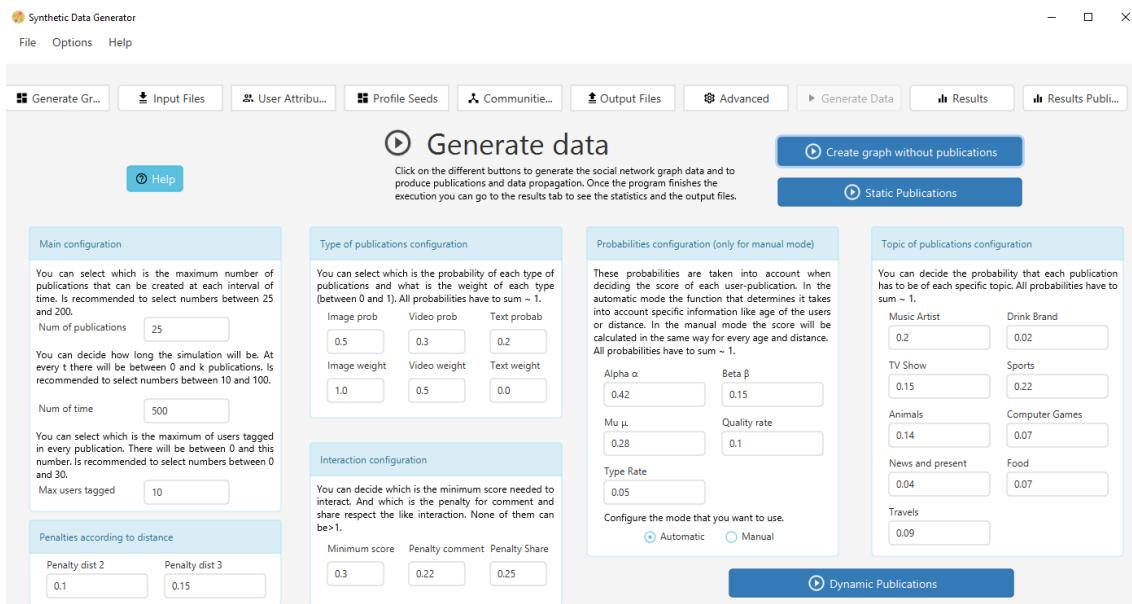
Ens permet modificar el percentatge de nodes llavors i la randomness de la generació del graf.



H. Interfície d'usuari - Advanced Settings

## i) Generate Data

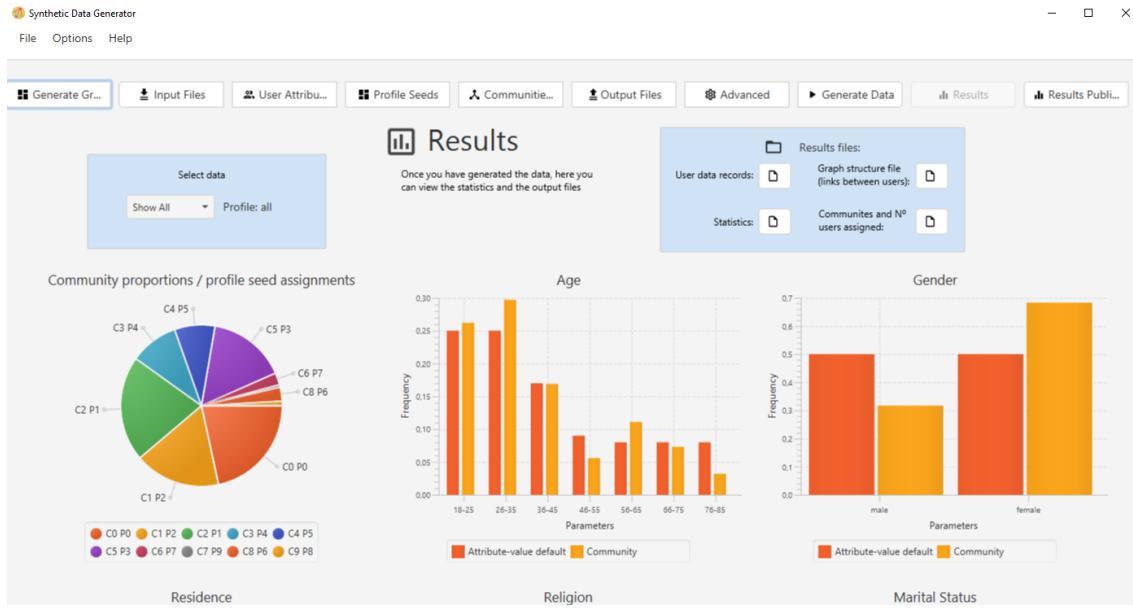
Com s'ha explicat a 4.3. *Pantalles de la interfície d'usuari*, aquesta pantalla ens permet configurar tots els paràmetres de la generació de les publicacions i de la propagació de la informació



H. Interfície d'usuari - Generate Data

## j) Results

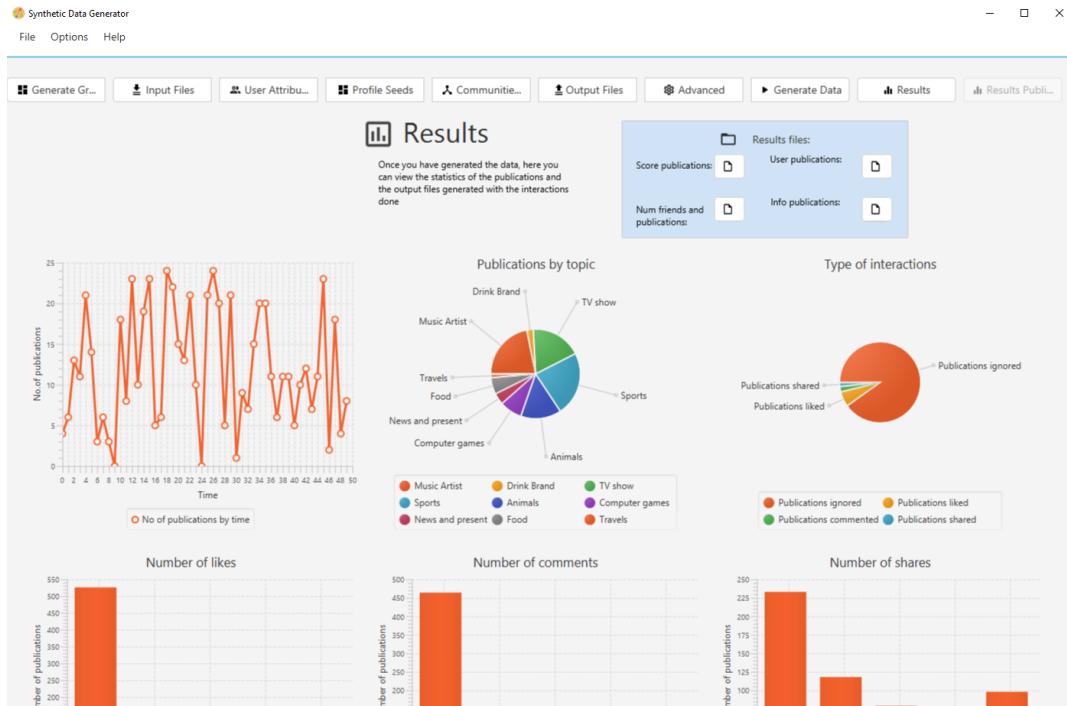
Permet visualitzar els resultats dels usuaris obtinguts.



H. Interfície d'usuari – Results

## k) Results Publications

Com s'ha explicat a 4.3. Pantalles de la interfície d'usuari ens permet veure un resum de la informació generada corresponent a les publicacions.



H. Interfície d'usuari - Results Publications

## Annex F: Més execucions del programa

### a) Execució 1

Num nodes	50.000
Num Edges	250.000
Max publications by time	25
Num time	500

Resta de paràmetres per defecte:



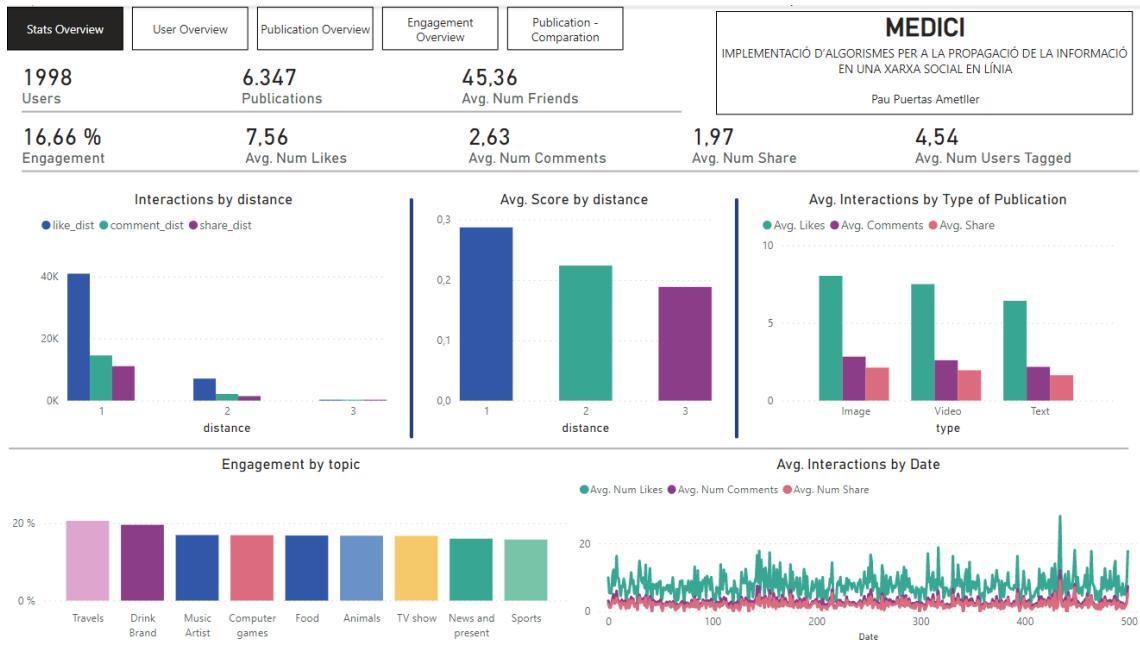
#### I. Execució 1

Veiem com l'engadgement és del 29,01%, molt elevat. Els usuaris tenen pocs amics (10 de mitjana) i 3,03 likes per publicació. Al tenir tant poques connexions, pràcticament totes les interaccions es produueixen a distància 1.

### b) Execució 2

Num nodes	2.000
Num Edges	50.000
Max publications by time	25
Num time	500

Resta de paràmetres per defecte:



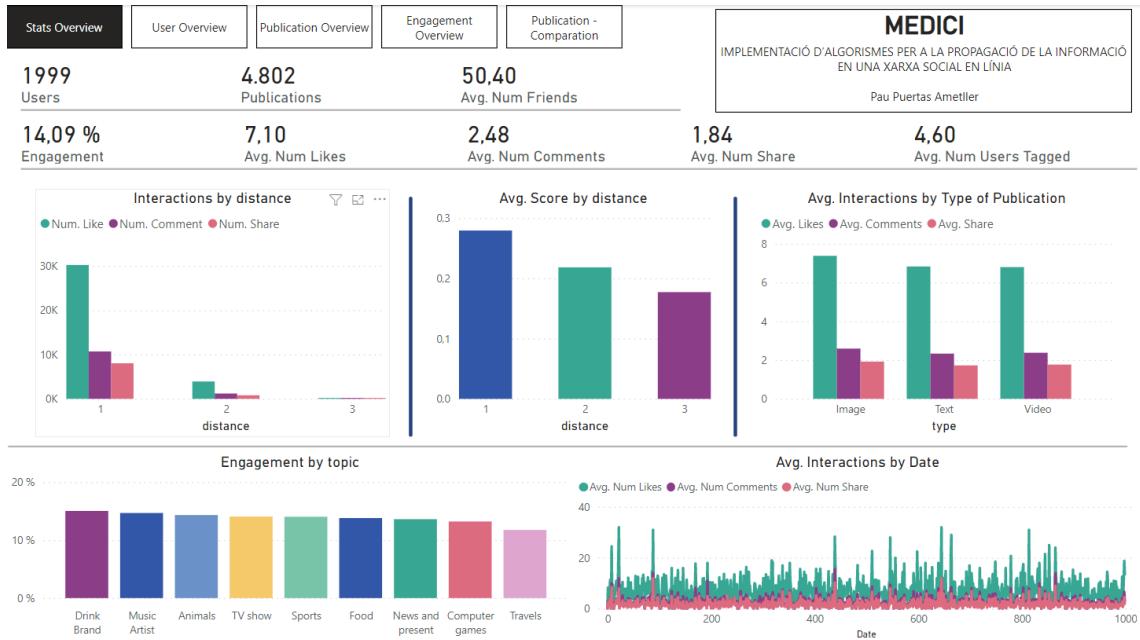
### I. Execució 2

Augmentem considerablement el nombre d'enllaços respecte els usuaris (mitjana d'amics 45,36) i aleshores l'engagement baixa fins al 16,66%, obtenint valors més propis de la realitat. Cada publicació té aproximadament 2 shares de mitjana.

### c) Execució 3

Num nodes	1.000
Num Edges	30.000
Max publications by time	10
Num time	1000

Resta de paràmetres per defecte:



### I. Execució 3

Augmentem encara més el nombre d'enllaços (50,40 amics de mitjana) i l'engagement es redueix encara més fins al 14,09%.

#### d) Execució 4

Num nodes	1.000
Num Edges	30.000
Max publications by time	25
Num time	200
Mode manual amb els paràmetres per defecte assignats	

Resta de paràmetres per defecte:



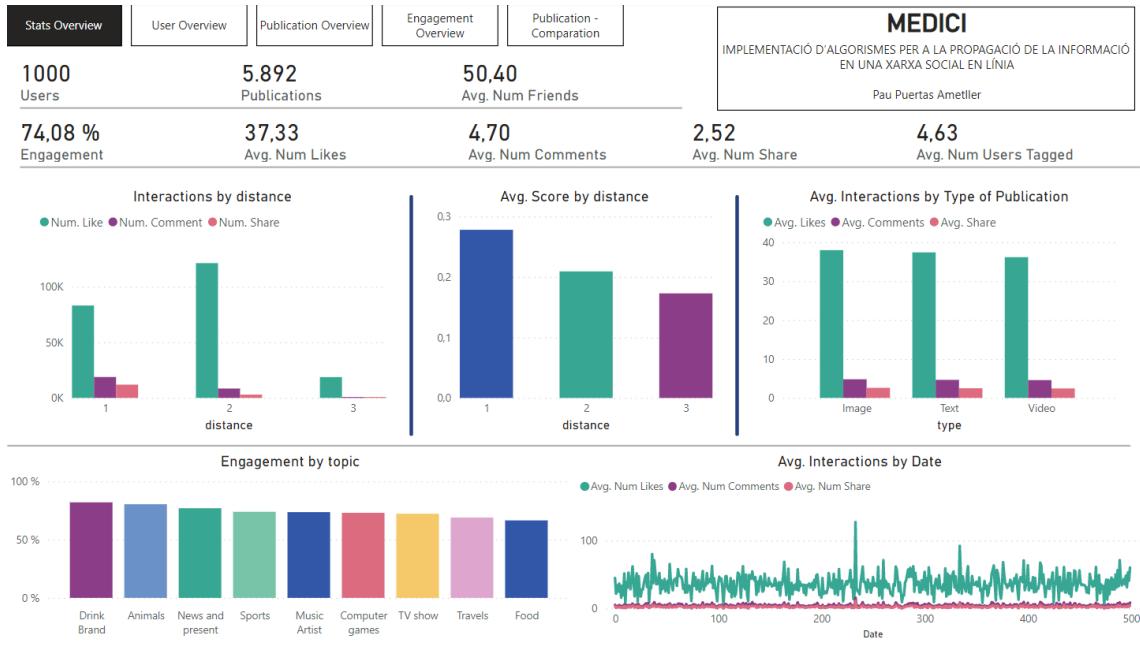
### I. Execució 4

Amb el mode manual veiem com l'engagement augmenta considerablement. Al no tenir-se en compte l'edat per tal de calcular l'score i utilitzar sempre els mateixos pesos per als paràmetres, s'interactua amb més freqüència.

### e) Execució 5

Num nodes	1.000
Num Edges	30.000
Max publications by time	25
Num time	500
Minimum Score	0

Resta de paràmetres per defecte:



### I. Execució 5

En aquest cas hem configurat el mínim score com a 0, permetent que qualsevol puntuació sigui capaç d'interaccionar (respectant la probabilitat). En aquest cas hi ha un engadiment del 74,08% desproporcionat totalment amb la realitat. Veiem com majoritàriament es produeixen likes i a distància 2 és quan més es produeixen.

### f) Execució 6

Num nodes	1.000
Num Edges	30.000
Max publications by time	25
Num time	500
Minimum Score	0,4

Resta de paràmetres per defecte:



### I. Execució 6

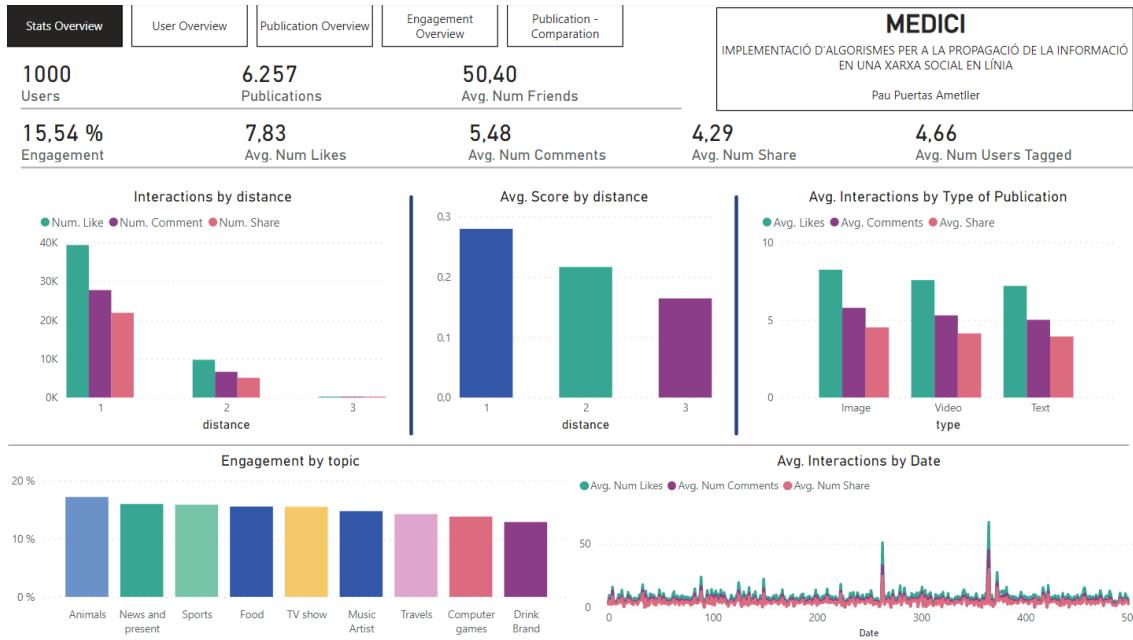
En aquest cas pasa totalment el contrari. Configurem el mínim score a 0,4 i veiem com només s'obte un engadgement del 0,50%.

Per tant queda demostrar que s'ha de trobar l'equilibri amb el mínim score per tal de garantir un engadgement adequat. Aquest paràmetre és un dels més importants.

### g) Execució 7

Num nodes	1.000
Num Edges	30.000
Max publications by time	25
Num time	500
Penalty Comment	0,1
Penalty Share	0,15

Resta de paràmetres per defecte:



### I. Execució 7

En aquest exemple s'ha baixat el penalty comment i el penalty share respecte els valors per defecte. D'aquesta manera, tot i que l'engagement no puja gaire, es produeixen molts més comentaris i es comparteixen més les publicacions.

Veiem com hi ha dos instants de temps on augmenta significativament les interaccions de manera molt particular.