Métodos Analíticos

Trabajo Final

Detección de comunidades en red de Twitter

Mario Becerra 124362
Karen Poblete 116452

Mayo 2015

Resumen

Se realizó el análisis de una red de *retweets* y *replies* entre usuarios relacionados al medio informativo mexicano. Se calcularon el grado de entrada y salida, así como el grado total de cada nodo. Además se calculó el *betweenness-centrality* de cada uno de ellos para encontrar la relevancia de cada nodo en la red. Se realizó la división de la red en comunidades por medio de un método rápido *greedy* que se centra en optimizar la modularidad de los nodos. Los resultados mostraron a AristeguiOnline como el usuario con mayor número de *retweets*, seguido por Pajaropolitico y El_Universal_Mx. Se encontraron 31 comunidades las cuales están asociadas a algún usuario importante, como los antes mencionados.

Keywords: Red social, Gephi, grafos, *betweenness-centrality*, *degree*, *in-degree*, *out-degree*.

1. Introducción

El estudio de redes ha sido de gran interés en los últimos años, debido a que representan un conjunto de actores y las relaciones entre ellos de una manera intuitiva y se puede tener una representación visual. Muchos sistemas usados diariamente pueden ser modelados por medio de redes, como la relación entre las páginas de internet, redes de transporte y servicios diversos, redes de parentescos y *Facebook*. Las redes comúnmente son modeladas por medio de grafos, ya sean direccionales y no direccionales.

El análisis de los grafos generados a partir de distintas redes, pueden ser utilizados para encontrar características particulares en las redes. Por ejemplo, encontrar las páginas web más importantes sobre algún tema en internet, definir comunidades en un grafo social como *Facebook* o *Twitter*, o incluso en una red telefónica; o calcular el camino más corto de un punto a otro en una red de transporte.

Este trabajo se centra en las redes sociales, las cuales tienen ciertas características que deben cumplir para ser catalogadas como tal [1]. Las principales son:

- Existe una colección de entidades que participan en la red, que comúnmente son personas.
- Existe por lo menos una relación entre las entidades de la red.
- Se asume que la localidad de los nodos no es aleatoria. Se debe entender por localidad la posición que ocupan los nodos con relación a los demás, comúnmente se tienden a juntar más con nodos que comparten características similares.

Un claro ejemplo de red social es *Facebook* que se puede representar como un grafo donde los usuarios son los nodos y sus conexiones son las relaciones de amistad entre ellos. De una red con éstas características se puede obtener información de las comunidades que lo integran, que nodos son más relevantes y la similitud entre ellos.

Se analizó una red social de *Twitter*, creada a partir de *retweets* y *replies*. El objetivo es encontrar comunidades a partir de las conexiones que existen entre los usuarios.

La información generada del análisis de redes de este tipo ayuda a entender el comportamiento de las entidades que lo conforman y en el caso de medios como la mercadotecnia y el comercio, pueden ser utilizadas para obtener alguna ventaja.

1.1. Comunidades en las redes sociales

Podemos definir comunidad como un conjunto de entidades que tienen alguna atributo en común y por eso tienen a juntarse en el mismo grupo. Aplicar métodos como *k-means* no capta la esencia de este tipo de redes, ya que cada entidad se asigna sólo a una comunidad y nada más. Las entidades que forman parte de redes sociales comúnmente forman parte de más de una comunidad. El análisis de una red social difiere de las demás redes en que las entidades pueden formar parte de diferentes comunidades, por ejemplo una persona puede ser parte de una institución, un grupo de amigos, pero también pertenece a una familia.

2. Descripción de los datos

Como se mencionó antes, la red con la que se trabajó fue creada a partir de *retweets* y *replies* de *tweets* relacionados al medio informativo mexicano, recolectados el 14 de mayo de 2015. Es una red dirigida con $38,423$ nodos y $60,943$ arcos, en la cual los nodos representan a usuarios, y un arco va del nodo A al nodo B si A *retweeteó* a B. Ejemplo, si *@AristeguiOnline* publica un *tweet* y *@sopitas* lo *retweeteó*, entonces hay un arco que va de *@sopitas* a *@AristeguiOnline*. Además, los arcos tienen peso, este peso se asigna de acuerdo al número de veces que un usuario *retweeteó* a otro. Por ejemplo, en el caso de *@sopitas* y *@AristeguiOnline*, si *@sopitas* solo *retweeteó* un *tweet* entonces el peso del arco es 1, pero si *retweeteó* 3 veces (no necesariamente los mismos *tweets*), entonces el peso del arco es 3.

Contamos con dos archivos, uno donde se describen los nodos que existen en el grafo (network_rich_nodes.csv), los cuales tienen los siguientes campos:

- *id*. El identificador del nodo.
- *user-name*. Nombre del usuario.
- *followers*. La gente que el usuario representado por el nodo sigue.
- *friends*. El número de amigos que tiene el usuario.
- *location*. La localización del usuario.
- *impacto*. Nos indica la importancia del nodo en la red.

El archivo network_edges_Prueba-Medios.csv describe las aristas del grafo. El grafo es direccional y por lo tanto las aristas tienen un origen y un destino como se había mencionado con anterioridad.

- *source*. Nodo origen de la arista.
- *target*. Nodo destino de la arista.
- *weight*. Peso de la arista, relacionado con el número de veces que un usuario hace *retweet* o *reply* al nodo destino.
- *rel-type*. El tipo de relación que existe entre los nodos (RT o *reply*.
- *topic*. El tema del *tweet* que fue *retweeteado* o respondido.

Los usuarios con mayor número de *followers* en la red se muestran en la tabla 1. Se muestra también el número de *friends*, esto es, el número de usuarios a los que sigue.

	id	*followers*	*friends*
1	CarlosLoret	5055942	762
2	lopezdoriga	4957223	1024
3	AristeguiOnline	3983364	6
4	Adela_Micha	3617556	320
5	aristeguicnn	3415161	19
6	brozoxmiswebs	3135718	697
7	El_Universal_Mx	3009308	11408
8	revistaproceso	2740162	3930

Cuadro 1: Usuarios con mayor número de *followers*

3. Metodología

El principal objetivo es identificar comunidades en el grafo. Existen varios tipos de algoritmos de detección de comunidades, algunos son divisivos, en el sentido que detectan ligas inter-comunidad y después los quitan de la red; otros son aglomerativos, que van juntando nodos recursivamente; y otros están basados en la maximización de una función objetivo.

3.1. Modularidad

Una medida muy usada para hacer conglomerados en redes sociales es la modularidad, la cual es un escalar entre -1 y 1 que mide la fuerza de la división de una red en conglomerados. Una red con modularidad alta tiene conexiones fuertes dentro de las comunidades, pero débiles entre las comunidades. Muchos de los algoritmos basados en optimización como los mencionados anteriormente buscan encontrar particiones que maximicen la modularidad.

En particular, la modularidad Q está definida como la fracción de arcos que están dentro de cada comunidad menos el número esperado de arcos en cada comunidad de un grafo aleatorio con la misma distribución de grados de entrada y salida que el grafo que se estudia. Matemáticamente, esto se ve como[2]

$$Q = \frac{1}{2m}\sum_{i,j}(A_{ij} - \frac{k_i k_j}{2m})\delta(c_i, c_j), \tag{1}$$

donde A_{ij} representa el peso del arco entre los nodos i y j, $k_i = \sum_j A_{ij}$ es la suma de los pesos de los arcos que salen del nodo i, c_i es la comunidad que se le asigna al vértice i, $\delta(u, v)$ es 1 si $u = v$ y 0 en otro caso, y $m = \frac{1}{2}\sum_{ij} A_{ij}$.

Una desventaja de utilizar la modularidad como función objetivo, es que puede fallar en encontrar comunidades pequeñas en una red muy grande, esto debido a la naturaleza de la función de modularidad, que resta el número esperado de vértices en una red aleatoria, el cual va disminuyendo mientras la red va creciendo; por lo que esto puede ser menor que uno, entonces la modularidad puede interpretar esto como signo de correlación fuerte entre dos comunidades, por lo que las juntaría en una sola comunidad.

3.2. *Betweenness*

Betweenness o intermediación es una medida del número de veces que un nodo actúa como puente en el camino más corto entre dos nodos. Es una forma de cuantificar el control que tiene un nodo en la comunicación existente entre otros. La idea básica detrás de esta medida es que los nodos con mayor *betweenness* son los que aparecen con mayor probabilidad en los caminos más cortos, de esta forma, es una medida de centralidad en una red. Formalmente se puede definir como [3],

$$C_{BET}(i) = \sum_{j,k} \frac{b_{jik}}{b_{jk}} \tag{2}$$

donde b_{jk} es el número de caminos más cortos desde el nodo j hasta el nodo k, y b_{jik} el número de caminos más cortos desde j hasta k que pasan a través del nodo i.

Los nodos con un alto valor de intermediación son muy importantes en la estructura de una red, ya que comunican comunidades con otras. Comúnmente los valores más altos de *betweenness* son obtenidos por los nodos que están en los bordes de las comunidades. Si uno nodo con alta intermediación desaparece, las comunidades podrían quedar incomunicadas. Calcular el *betweenness* resulta ser una tarea complicada y tardada, pues se recorre toda la red nodo por nodo.

Esta noción de centralidad en los nodos se puede extender a las aristas, y de esta forma el *betweenness* de una arista es el número de caminos más cortos entre el par de nodos que corren a través de esta arista. Así, las aristas que conectan comunidades tendrán mayor nivel de *betweenness*, pues al quitar estas aristas, las comunidades quedarían separadas una de la otra. Esta noción de *betweenness* de aristas se puede explotar para encontrar comunidades en la red. El algoritmo Girvan-Newman hace esto siguiendo los siguientes pasos:

1. Se calcula el *betweenness* de cada arco
2. Se quita el arco con mayor *betweenness*
3. Se recalcula el *betweenness* de los arcos afectados por la acción de haber quitado el arco
4. Se repiten los pasos 2 y 3 hasta que no queden más arcos

Este algoritmo devuelve un dendrograma en el cual las hojas son los nodos, con esto se pueden asignar comunidades a los nodos.

3.3. Gephi

Gephi es un analizador y visualizador abierto de redes escrito en *Java* sobre la plataforma de *NetBeans*[4]. Fue inicialmente desarrollado por estudiantes de la Universidad de Tecnología de Compiègne (UTC) en Francia. *Gephi* cuenta con un conjunto de herramientas adecuados para el análisis de redes de diferentes tipos.

Se utilizó *Gephi* como herramienta de visualización para el análisis de la red de *tweets* descrita con anterioridad.

3.4. igraph

Otra herramienta que se utilizó para este trabajo fue *igraph* [5] a través de R. Se usó para hacer el cálculo de varios atributos del grafo: degree, in-degree, out-degree, betweeness-centrality, para los nodos y edge-betweeness, para las aristas.

También se encontraron comunidades en el grafo maximizando la modularidad con un algoritmo utilizando la función *fastgreedy.community*. Dicho método es una aproximación jerárquica que pretende maximizar la modularidad de una manera *greedy*. Al inicio todos los nodos están separados y con las iteraciones se empiezan a juntar en comunidades hasta que ya no se puede optimizar más la modularidad. Comúnmente es la primera aproximación que se usa, ya que se configura fácilmente y es rápido.

4. Resultados

4.1. Grado de los nodos y centralidad

Se realizó el cálculo del *grado* (*degree*), *grado de entrada* (*in-degree*) y *grado de salida* (*out-degree*). El grado de entrada y el grado de salida de un nodo son el número de aristas que entran y salen de este, mientras que el grado indica el número de aristas totales que entran y salen de él, es decir, la suma del grado de entrada y del grado de salida.

	Promedio	Desv. Est.	Min	Max
Degree	55.72	3.17	1	5076
In-degree	1.54	1.586	0	5076
Out-degree	55.72	1.586	0	29

Cuadro 2: Grado de los nodos

La tabla 2 muestra los valores mínimos, máximos y en promedio de los grados de los nodos, así como la desviación estándar. El *in-degree* máximo es 5076, esto significa que por lo menos un usuario fue *retweeteado* o respondido por 5076 usuarios distintos. La tabla 3 muestra los cinco usuarios con mayor *in-degree*, *out-degree* y *degree*, aquí se ve que el usuario que más fue *retweeteado* o respondido fue *AristeguiOnline*, seguido de *Pajaropolitico* y de *El_Universal_Mx*.

El *out-degree* no dice nada muy interesante, pues son simplemente usuarios que *retweetearon* a muchos usuarios distintos.

	Id_Usuario	In-degree	Id_Usuario	Out-degree	Id_Usuario	Degree
1	AristeguiOnline	5076	EverardoSosaC	29	AristeguiOnline	5076
2	Pajaropolitico	3229	HanniKassam	28	Pajaropolitico	3229
3	El_Universal_Mx	2952	Tecpanero	25	El_Universal_Mx	2952
4	DeniseDresserG	2779	Tzirintzi	23	DeniseDresserG	2781
5	SinEmbargoMX	2535	Eloisavizzuett	23	SinEmbargoMX	2536

Cuadro 3: Tabla del grado de los nodos

También se realizó el cálculo de *betweenness-centrality* de cada nodo. Como se mencionó antes, altos valores de *betweenness* implica alta importancia del nodo en la red, pues al quitar dichos nodos la red se puede fragmentar. La tabla 4 muestra los nodos con mayor *betweenness-centrality*. Puede verse que Televisa abarca tres de 5 nodos con mayor centralidad.

no.	Usuario	*betweenness-centrality*
1	NTelevisa_com	8797.167
2	DeniseDresserG	4120.333
3	Foro_TV	3950.833
4	CarlosLoret	3643.167
5	Milenio	2419.000

Cuadro 4: Tabla de los 5 mayores *betweenness-centrality*

4.2. Búsqueda de comunidades

Optimizando modularidad de forma *greedy* se encontraron 31 comunidades, las cuales tienen diferentes tamaños, definidos en la tabla 5.

En la figura 1 se muestran la red completa con los nodos coloreados de acuerdo a las comunidades encontradas. Las etiquetas de los nodos son proporcionales al valor de *in-degree* de cada nodo, es decir, los más *retweeteados* son más grandes, como *Aristegui_Online*, *DeniseDresserG* y *El_Universal_Mx*.

Com	1	2	3	4	5	6	7	8	9	10	11
Tam	1103	996	890	1065	1136	1685	962	1127	1557	997	1455
%	2.87	2.59	2.32	2.77	2.96	4.39	2.5	2.93	4.05	2.59	3.25
Com	12	13	14	15	16	17	18	19	20	21	22
Tam	1468	1534	2201	1823	386	2690	769	695	2466	1664	1507
%	3.97	3.99	5.73	4.74	1	7	2	1.81	6.42	4.33	3.82
Com	23	24	25	26	27	28	29	30	31		
Tam	2105	2198	374	2551	193	241	359	223	2		
%	5.48	5.72	0.97	6.64	0.5	0.63	0.93	0.58	0.01		

Cuadro 5: Tamaños por comunidad

Figura 1: Identificación de comunidades

Utilizando *Gephi* se pudo hacer un análisis de cada una de las comunidades que integran la red. A continuación se muestran algunos ejemplos de comunidades encontradas. La figura 2 muestra la comunidad en la cual *SinEmbargoMX* es el usuario más importante o central; la figura 3 muestra la comunidad que se centra en *lopezdoriga*. Las demás comunidades encontradas se pueden ver en la sección de Anexos.

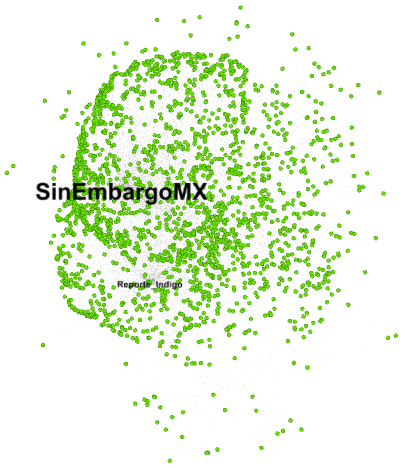

Figura 2: Comunidad relacionada con SinEmbargoMx

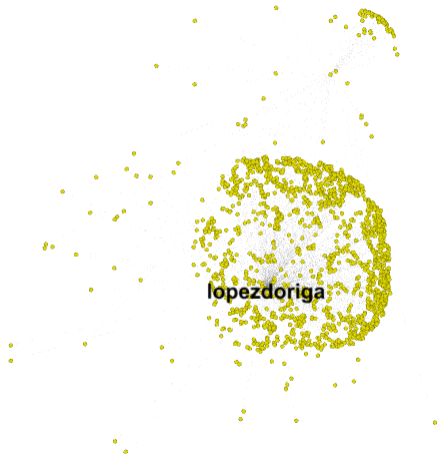

Figura 3: Comunidad relacionada con lopezdoriga

5. Conclusiones y trabajo futuro

Se puede concluir que AristeguiOnline, Pajaropolitico, El_Universal_Mx, DeniseDresserG, y otros usuarios de Televisa son los usuarios con mayor influencia en el día que se analizó la red, ya que han mostrado ser los más *retweeteados*.

Las comunidades encontradas se centraban principalmente en usuarios que tienen muchos *followers* y/o que fueron *retweeteados* un número considerable de veces. Esto es natural debido a la naturaleza del algoritmo que se usó para encontrar comunidades.

Para ahondar más en este tema se podría analizar una red creada de la misma forma que esta, pero abarcando mayor número de días. Algo interesante sería que, una vez creadas las comunidades, se analice el contenido de los *tweets* de cada comunidad y ver si se encuentran parecidos.

Agradecimientos

Agradecemos a la empresa Sinnia y en particular a Guillermo Garduño por brindarnos su apoyo y proporcionarnos los datos utilizados en este trabajo.

Referencias

[1] Jure Leskovec, Anand Rajaraman, and Jeffrey David Ullman. *Mining of massive datasets*. Cambridge University Press, 2014.

[2] Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. 2008(10):P10008.

[3] Jimeng Sun and Jie Tang. A survey of models and algorithms for social influence analysis. pages 177–214, 2011.

[4] Mathieu Bastian, Sebastien Heymann, and Mathieu Jacomy. Gephi: An open source software for exploring and manipulating networks, 2009.

[5] Gabor Csardi and Tamas Nepusz. The igraph software package for complex network research. *InterJournal*, Complex Systems:1695, 2006.

[6] Renaud Lambiotte, J.-C. Delvenne, and Mauricio Barahona. Laplacian dynamics and multiscale modular structure in networks.

[7] Santo Fortunato and Marc Barthélemy. Resolution limit in community detection. 104(1):36–41.

Anexos

Código de R

```
## ------------------------------------------------------------------------
library(igraph)
library(dplyr)
edges <- read.csv("./Datos/14mayo/network_edges_Prueba-Medios.csv", header = TRUE, sep = ",")
nodes <- read.csv("./Datos/14mayo/network_nodes_Prueba-Medios.csv", header = TRUE, sep = ",", quote = "
f <- file('./redmedios.gml')
gr <- read.graph(f, format = 'gml')


## ------------------------------------------------------------------------
nodes2 <- nodes
nodes2$id <- gsub("\"", "", nodes2$X.id.)
nodes2$foll<- as.numeric(gsub("\"", "", nodes2$X.followers.))
nodes2$friends <- as.numeric(gsub("\"", "", nodes2$X.friends.))
nodes2$X.id. <- NULL
nodes2$X.followers. <- NULL
nodes2$X.friends. <- NULL
arrange(nodes2, desc(foll)) %>% head(20)
xtable(arrange(nodes2, desc(foll)) %>% head(8))

## ------------------------------------------------------------------------
```

```
grafo <- data.frame(edges[,1], edges[,2])
g <- graph.data.frame(grafo, directed=TRUE)
g <- simplify(g)

betweennessG <- betweenness(g)
indegreeG <- degree(g, mode="in")
outdegreeG <- degree(g, mode="out")
totaldegreeG <- degree(g)
inclosenessG <- closeness(g, mode='in', weight = edges$weight)
outclosenessG <- closeness(g, mode='out')
totalclosenessG <- closeness(g)

res <- data.frame(id = V(g)$name, betweennessG, indegreeG, outdegreeG, totaldegreeG, inclosenessG,outcl
write.table(res,file="nodosGraph.csv",sep=",")
arrange(res, desc(indegreeG)) %>% head
arrange(res, desc(outdegreeG)) %>% head

#Faltan estadísticas de pesos

#ebc <- edge.betweenness.community(g, directed=T)
#res2 <- data.frame(ebc)
#write.table(res,file="aristasGraph.csv",sep=",")


## ------------------------------------------------------------------------
nodes_means <-colMeans(res)
nodes_means
nodes_min <- apply(res, 2, min)
nodes_min
nodes_max <- apply(res,2,max)
nodes_max
nodes_sd <- apply(res,2,sd)
nodes_sd

alto_degree <- arrange(res, desc(degreeG))
head(alto_degree)
alto_indegree <- arrange(res, desc(indegreeG))
head(alto_indegree)
alto_outdegree <- arrange(res, desc(outdegreeG))
head(alto_outdegree)


## ------------------------------------------------------------------------
alto_betw <- arrange(res, desc(betweennessG))
head(alto_betw)

## ------------------------------------------------------------------------
g_un <- graph.data.frame(grafo, directed=FALSE)
g_un_s <- simplify(g_un)
comm <- fastgreedy.community(g_un_s, membership=TRUE, weights = edges$weight)
#comm <- edge.betweenness.community(g, weights = edges$weight, directed=TRUE)
```

```
#A qué comunidad pertenece cada nodo
membership(comm)

#Tamaño de cada comunidad
sizes(comm)

#Miembros de las comunidades
communities(comm)
res2 <- data.frame(res, com_mod=comm$membership)
write.table(res2,file="nodosGraph2.csv",sep=",", row.names=F)
V(g)$comm_mod <- membership(comm)
write.graph(g, file = './grafo2.graphml', format='graphml')
```

Grafos de comunidades encontradas

Figura 4: Comunidad relacionada con CNN México

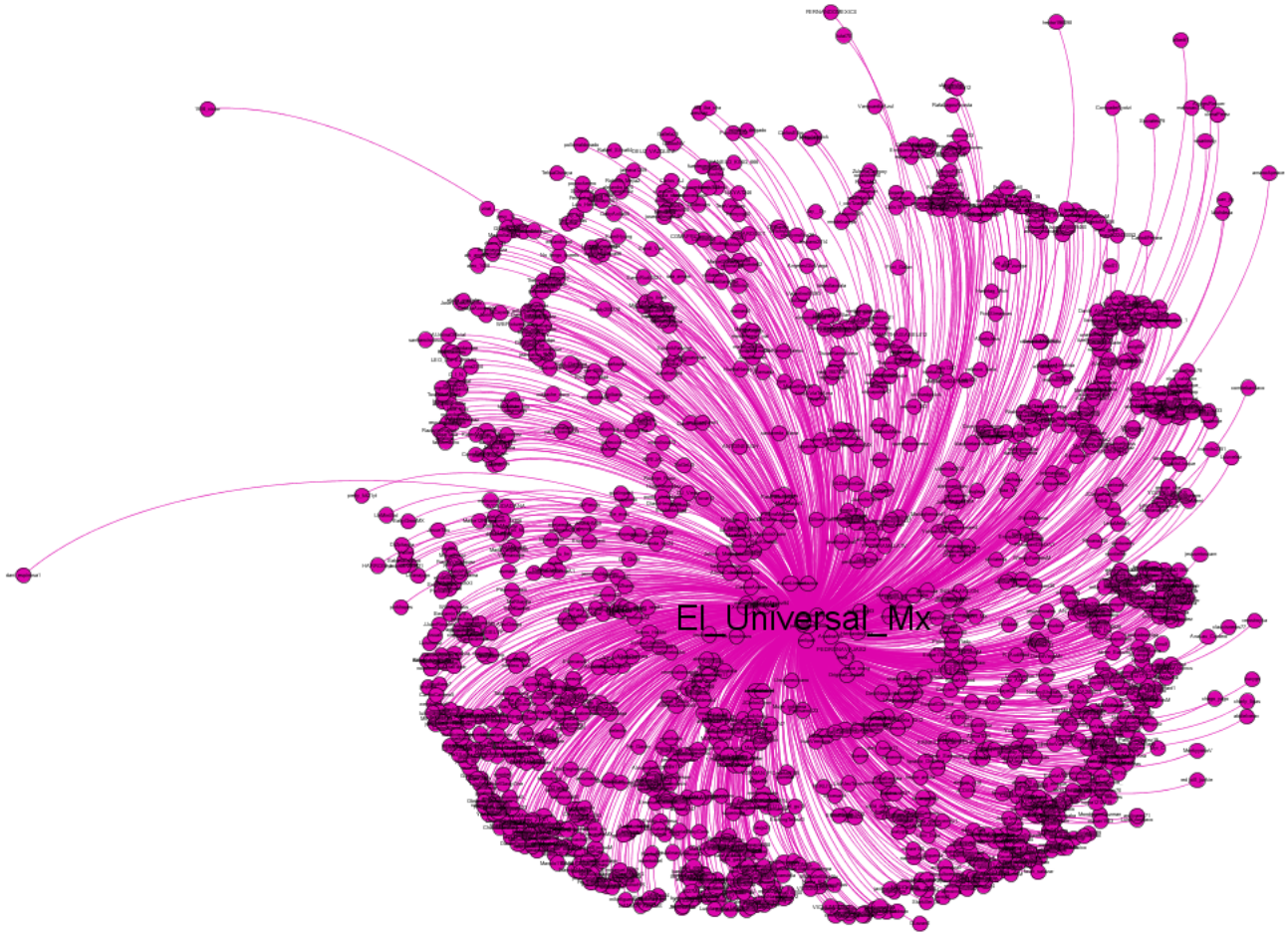

Figura 5: Comunidad relacionada con el periódico el universal

Figura 6: Comunidad relacionada con MX

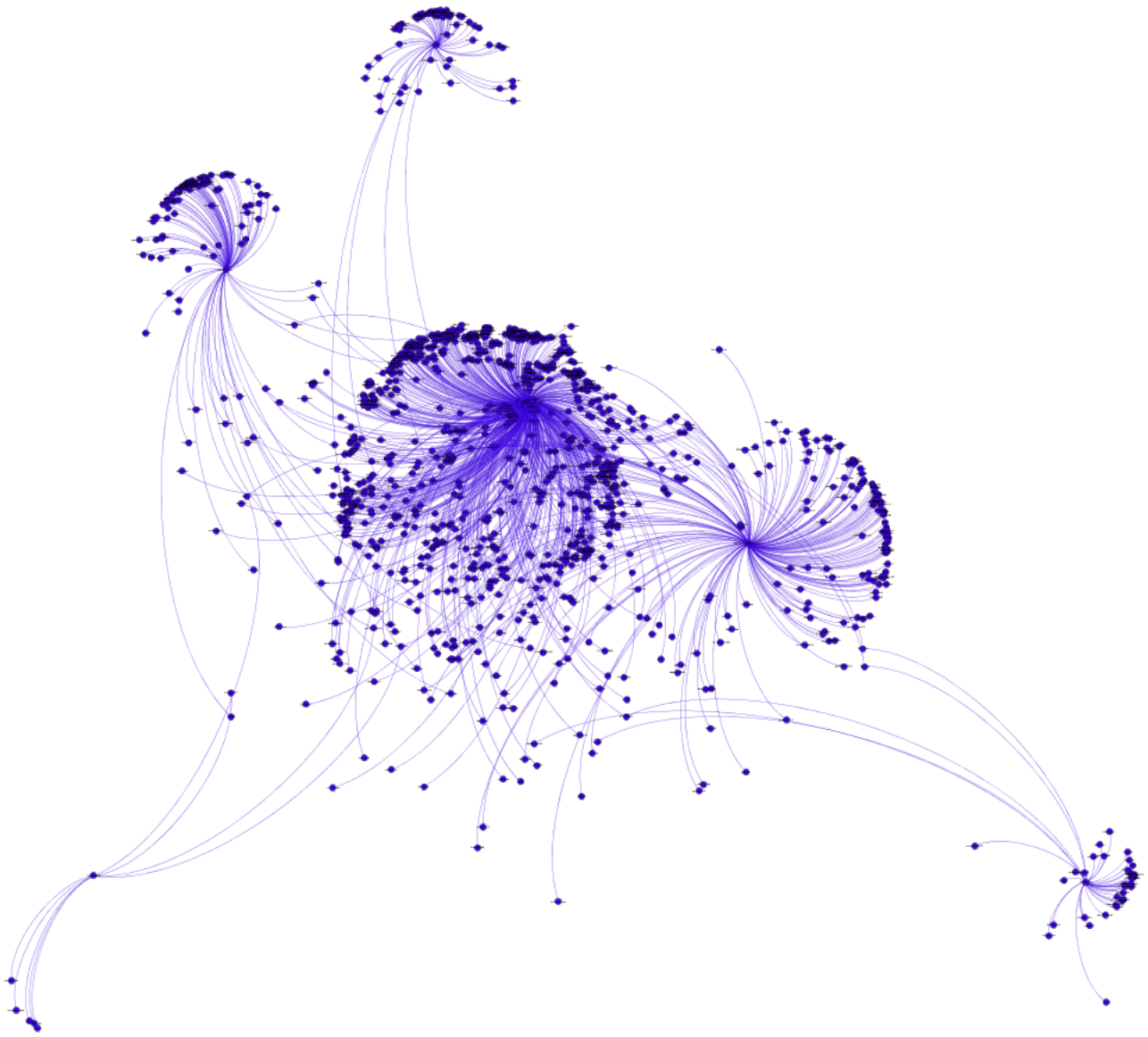

Figura 7: Comunidad relacionada con Foro TV

Figura 8: Comunidad relacionada con José Cardenas

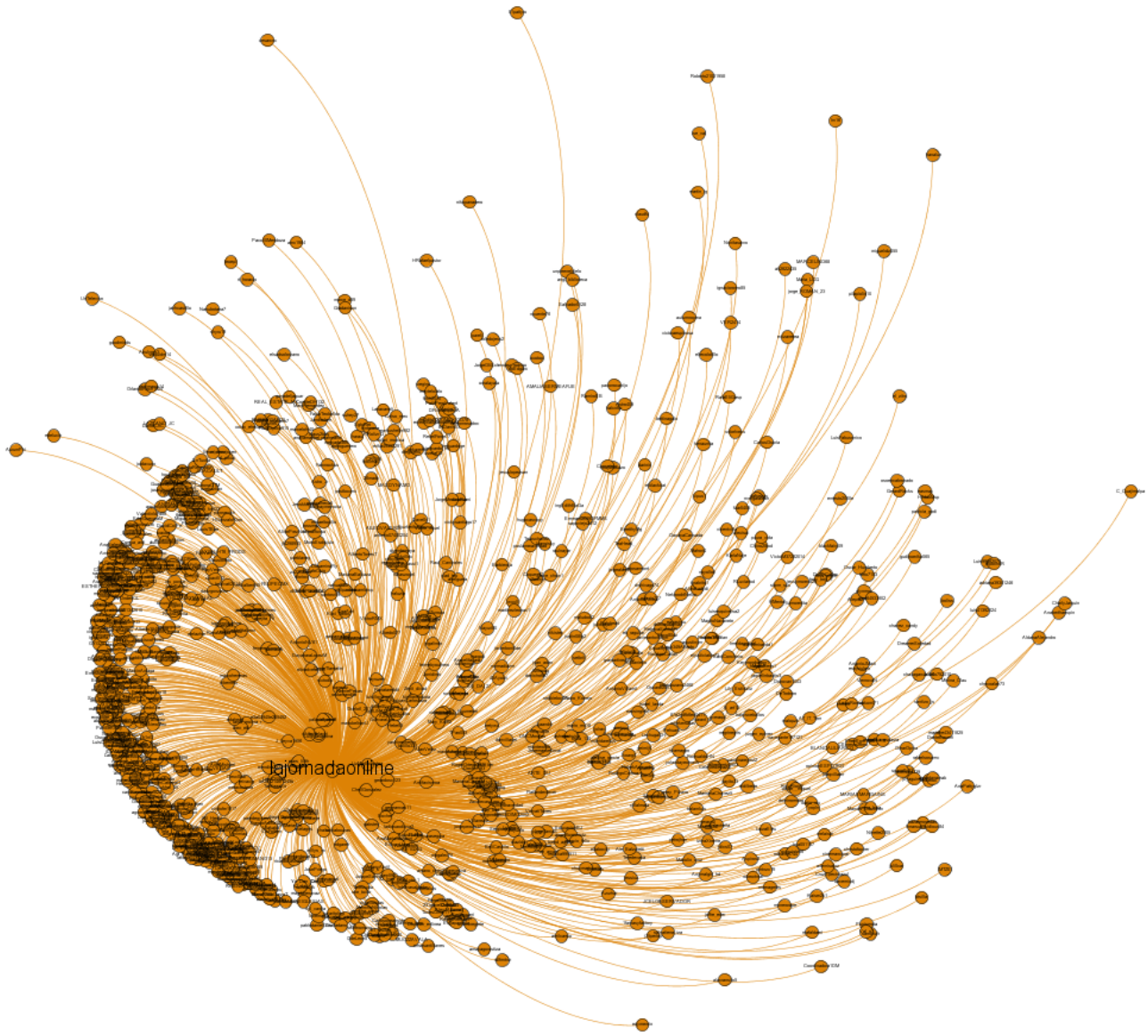

Figura 9: Comunidad relacionada con el periódico La Jornada

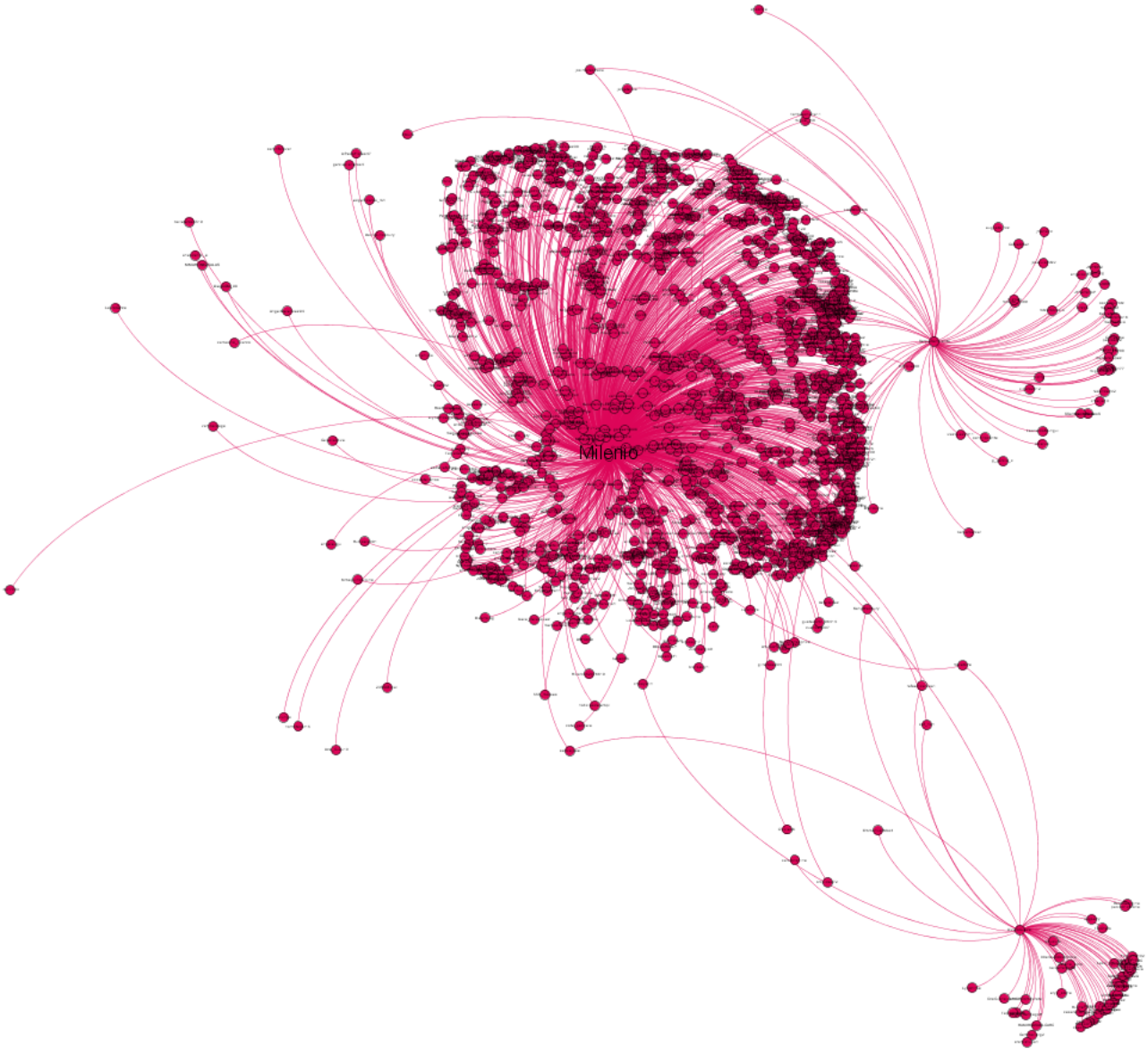

Figura 10: Comunidad relacionada con el periódico Milenio

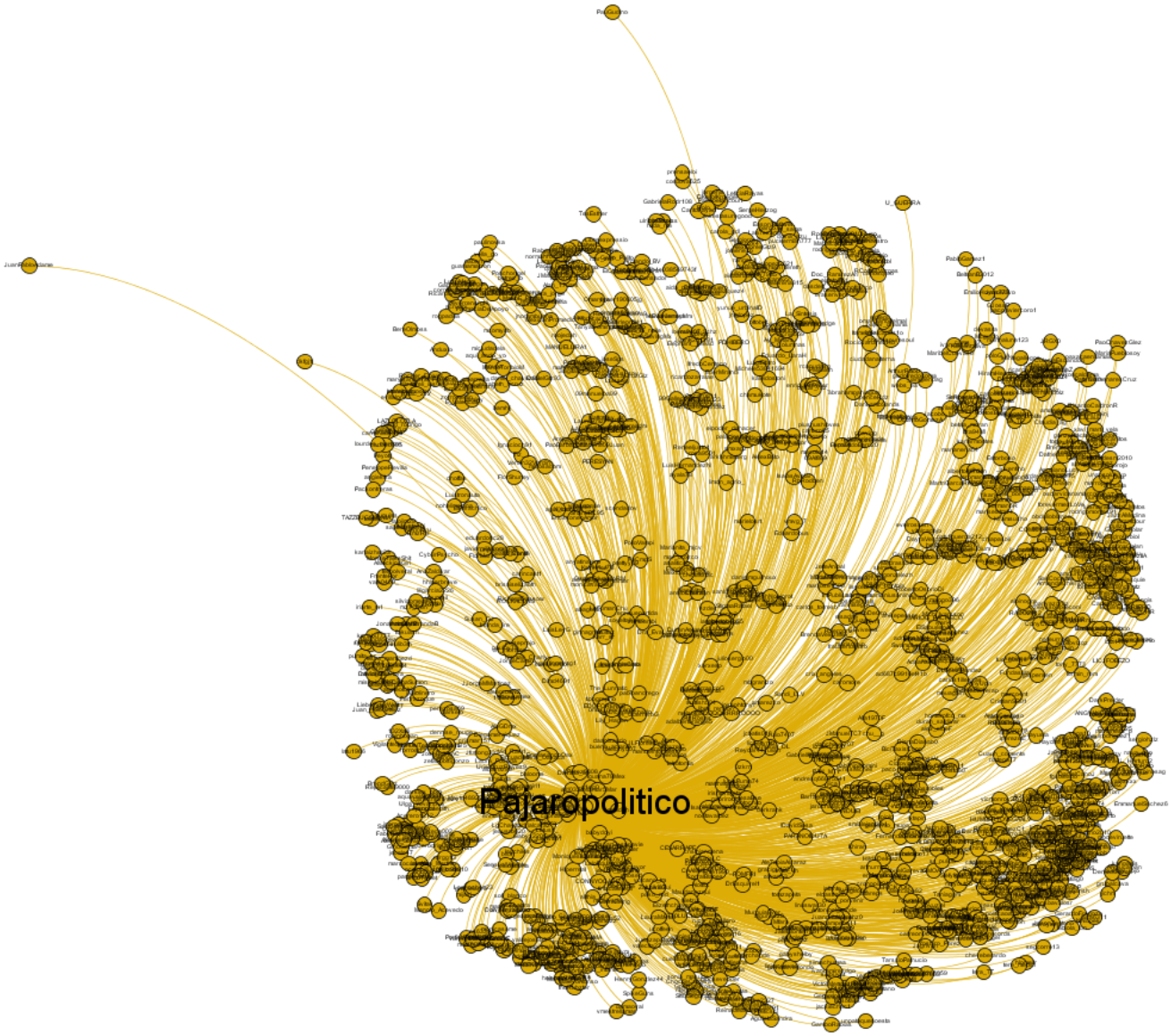

Figura 11: Comunidad relacionada con Pájaro Político

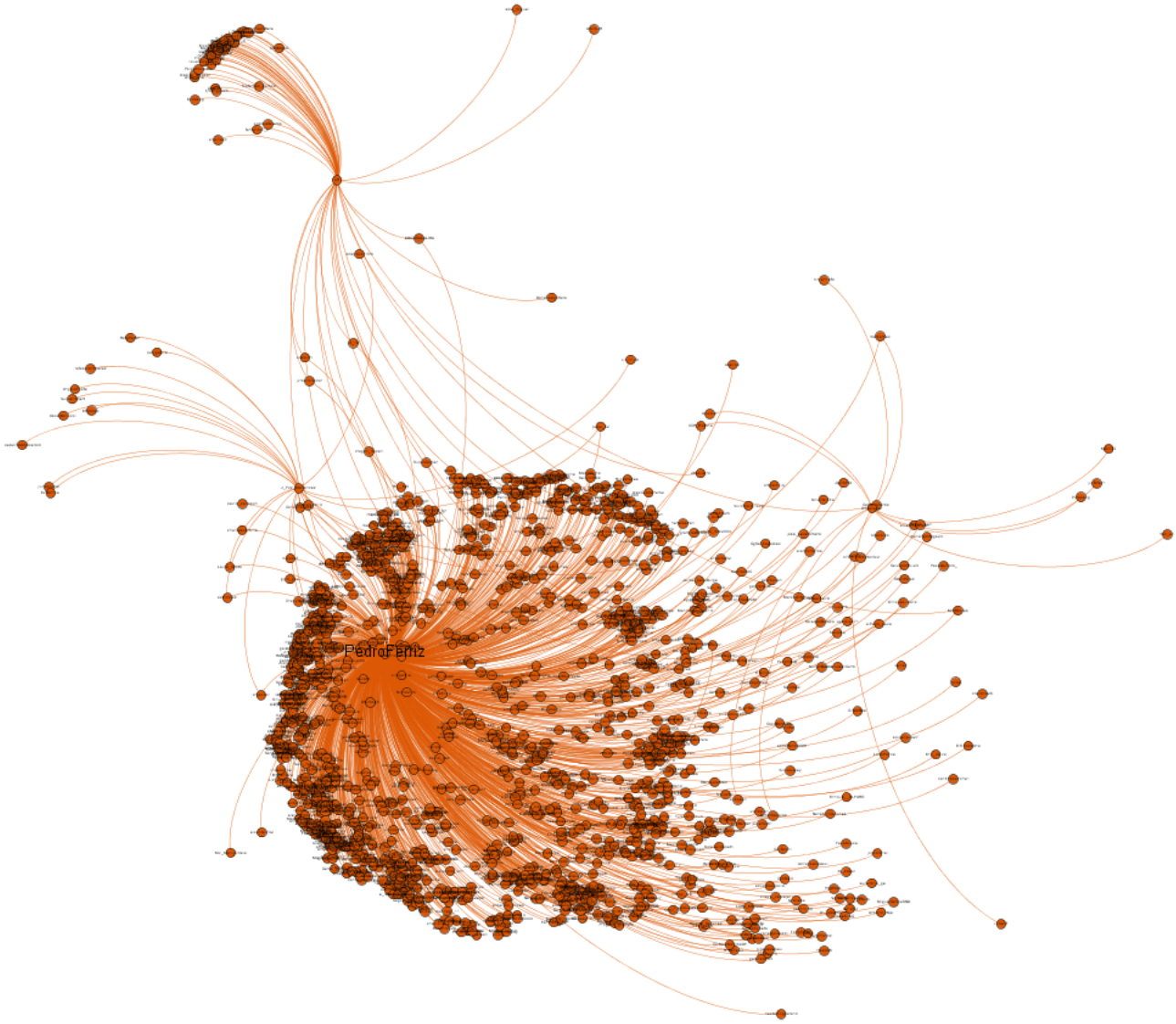

Figura 12: Comunidad relacionada con Ppedro Ferriz

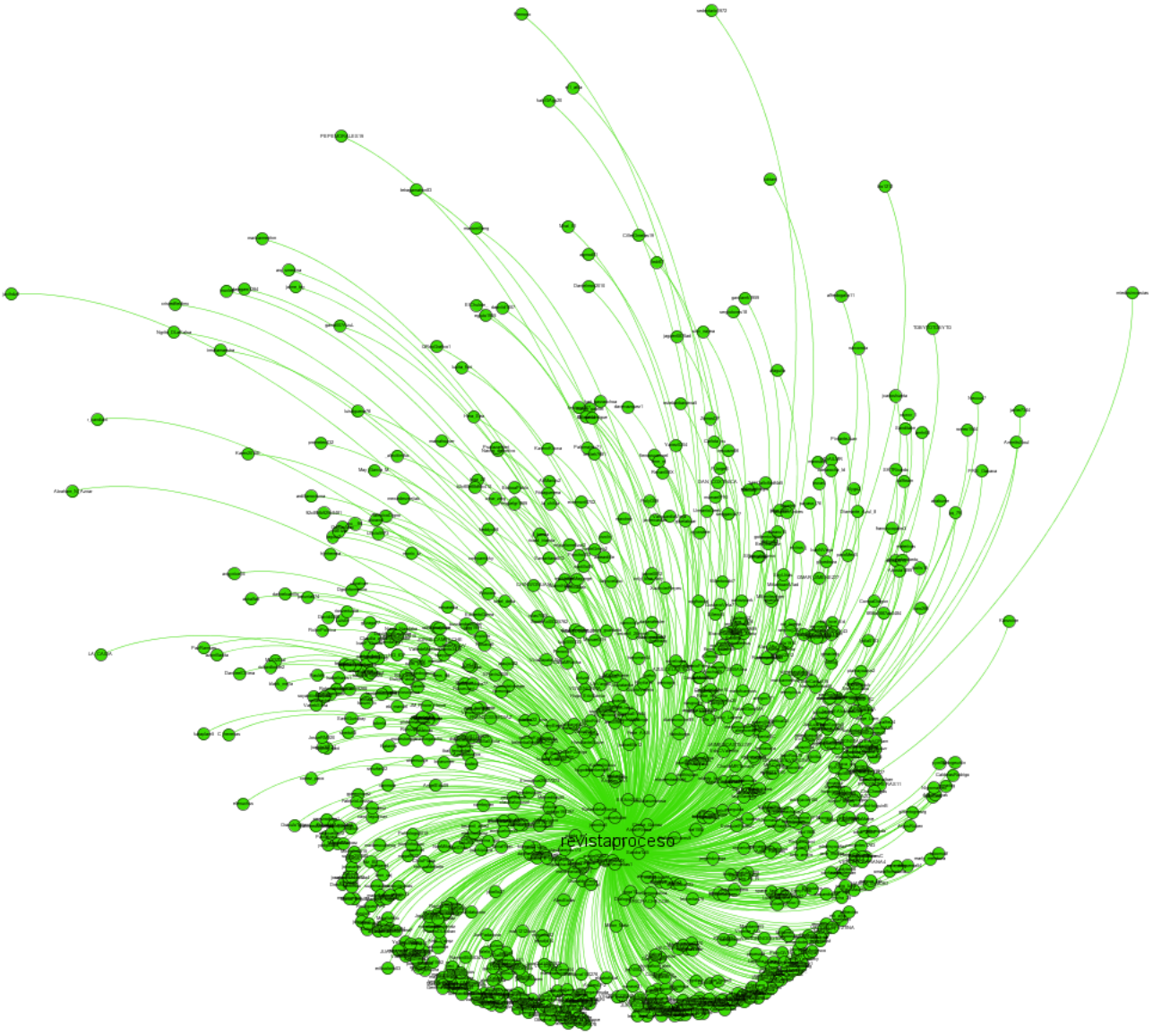

Figura 13: Comunidad relacionada con la revista Proceso

Figura 14: Comunidad relacionada con Sopitas