Facultad de Ciencias Económicas Empresariales (ICADE)

Visualización y Análisis de redes - Gephi

Autor: Baucells González, Javier
Director: Ramírez del Río, Antonio

Junio – 2024, Madrid

Resumen ejecutivo

El objetivo de este estudio consiste en el análisis de un set de datos de Twitter con *Gephi* extraído de un repositorio público, para ello, se procede a un tratado preliminar del dataset *'text_emotion.csv' en Python* para su correcta incorporación en *Gephi*. Este dataset contiene información referente a *'tweets'* publicados por usuarios en la red, así como el contenido de dicho tweet y el 'sentiment' que transmiten para facilitar el análisis de sentimientos.
Tras un análisis exploratorio preliminar junto a la ejecución de diferentes *'layouts'* se consigue la disposición deseada de los datos procediendo a realizar un estudio de influencias, comunidades y la distribución de los sentimientos en la muestra.
Una vez demostrada la metodología que es seguida por las representaciones visuales de la muestra, se estudian las zonas de mayor y menor conectividad entre las aristas. De esta manera, se procede a un análisis más exhaustivo de los grupos con mayor conectividad y de los 10 usuarios más mencionados en la red junto a las emociones que reciben.
Por último, se presentan los diferentes estadísticos extraídos con *Gephi*, así como las conclusiones obtenidas en el desarrollo del trabajo y recomendaciones para futuros estudios.

Palabras clave: *Datos Twitter, Análisis de redes, Sentiment Analysis, Redes Sociales, Visualización de datos, Gephi, Detección de comunidades.*

Abstract

The aim of this study is to analyze a Twitter dataset with Gephi extracted from a public repository. To do so, a preliminary processing of the dataset 'text_emotion.csv' in Python is carried out for its correct incorporation into Gephi. This dataset contains information about tweets posted by users on the network, as well as the content of the tweet and the sentiment they convey to facilitate sentiment analysis.
After a preliminary exploratory analysis together with the execution of different layouts, the desired arrangement of the data is achieved and a study of influences, communities, and the distribution of sentiment in the sample is carried out.
Once the methodology that is followed by the visual representations of the sample has been demonstrated, the areas of greater and lesser connectivity between the edges are studied. In this way, the procedure leads to a more exhaustive analysis of the groups with the highest connectivity and of the 10 most mentioned users in the network together with the emotions they receive.
Finally, the different statistics extracted with Gephi are presented, as well as the conclusions obtained in the development of the work and recommendations for future studies.

Keywords*: Twitter Data, Network Analysis, Sentiment Analysis, Social Networking, Data Visualisation, Gephi, Community Detection.*

INDICE

INDICE DE FIGURAS

1. Introducción y contextualización

El análisis de sentimiento en redes es una herramienta de *'text mining'* dentro del campo del procesamiento del lenguaje natural (NLP) que puede reportar significativos *insights* acerca de las opiniones y sentimientos de la población estudiada. Es un recurso de gran ayuda para la extracción de datos cualitativos de plataformas digitales con innumerables aplicaciones. Por ejemplo, para detectar segmentos de consumidores, la opinión respecto a una marca o conocer la intención de voto de un grupo de individuos.

Las redes sociales son una de las fuentes activas más grandes de información de nuestra sociedad, millones de datos son enviados, por ejemplo, en Twitter se envían 456.00 'tweets' por minuto, lo que hace un total de 656 millones de 'tweets' al día. Manejar estos grandes volúmenes de información sin realizar un estudio de estos sería desperdiciar recursos. Es por ello por lo que el 'sentiment analysis' es tan relevante y por lo que las empresas invierten tanto en procesos para sofisticar estos algoritmos, ya que, un 80% de los datos en nuestra sociedad son no estructurados, este tamaño, aumenta anualmente, por lo que, las herramientas para el análisis de este tipo de información cada vez serán más precisas y concretas.

Para el análisis de sentimiento, generalmente, se aplican algoritmos de *Machine Learning* que determinan qué tipo de sentimiento refleja un texto en base a unas reglas definidias previamente. Para ello, primero es necesario eliminar el ruido, proceder a la *tokenización* del texto, dividiéndolo en fragmentos o palabras para su procesamiento y normalización *(lematización y steaming).* La normalización se fundamenta en estas dos técnicas:
Lematización: consiste en una técnica para extraer los lexemas fundamentales de las palabras, utilizando como fuente un diccionario que contiene diferentes lexemas con la categoría que le corresponde, de esta forma, los algoritmos de *'text mining'* comprenden con mayor precisión el texto que están analizando.

Steaming: consiste en 'podar' las palabras, deduciendo sus prefijos y sufijos para conseguir acceder a su raíz, de esta forma, se reduce la cantidad de palabras en el *'corpus'* que será procesado por el algoritmo ejecutado.

Además, es necesario tener en cuenta cuestiones como la ambigüedad, el sarcasmo, la doble negación o el contexto que existe para no atribuir un sentimiento u opinión erróneo, especialmente, en las redes sociales, lugar en el que estos aspectos están a la orden del día debido a la alta variedad de opiniones y cantidad de usuarios en estas plataformas.

Este trabajo, tiene como objetivo principal, utilizar *Gephi* (software de código abierto para la visualización y análisis de redes), para analizar un *dataset* con análisis de sentimiento en Twitter. A lo largo del mismo, se exponen los objetivos generales y específicos que se desean alcanzar, la metodología empleada en el trabajo y los pasos previos de tratamiento de datos con Python para obtener los nodos y aristas que serán introducidos en *Gephi* con el objetivo de obtener una visualización adecuada que permita el estudio de la muestra.

Posteriormente, se exponen los estadísticos fundamentales de la red, que son comentados, atendiendo a las conclusiones obtenidas de sus valores, sirviendo como principales 'drivers' para futuras investigaciones.
Finalmente, se presentan las conclusiones generales y personales a la vez que recomendaciones para futuros estudios.

2. Objetivos del estudio

El objetivo principal del trabajo consiste en utilizar la herramienta de software libre *Gephi* para realizar un *'sentiment analysis'* a través de visualizaciones en *Gephi*. A su vez, existen una serie de objetivos específicos a conseguir para satisfacer todas las necesidades del estudio:

-OE 1: Encontrar una fuente fiable de datos en un repositorio online
-OE 2: Proceder al correcto tratado de los datos en Python
-OE 3: Visualizar correctamente la muestra
-OE 4: Disponer la visualización de modo que se puedan encontrar insights
-OE 5: Conectar la API de Twitter con Gephi
-OE 6: Realizar visualizaciones con datos extraídos de la API de Twitter
-OE 7: Estudio de comunidades
-OE 8: Análisis de los 10 individuos con mayor influencia en la red
-OE 8: Generación de estadísticos
-OE 9: Obtención de conclusiones y recomendaciones
-OE 10: Relacionar el estudio con 'Legatum'

Los objetivos de mayor complejidad consisten en la adecuada incorporación de los datos a la herramienta de visualización y el estudio en profundidad de las comunidades. Por otra parte, es necesario tener en cuenta que hay factores externos que pueden impedir la consecución de alguno de los objetivos específicos.
Al término del estudio, se revisarán todos los objetivos para comprobar si han sido cumplidos y determinar qué factores han impedido la consecución de otros. En el caso de no haber conseguido alguno de los objetivos, se propondrá un camino a tomar para futuros estudios de forma que esa cuestión no se encuentre indeterminada.

3. Metodología

En cuanto a la metodología aplicada para la realización de este TFG, se propone un razonamiento inductivo, partiendo de premisas particulares para llegar a conclusiones generales, especialmente en el estudio de la población, se esclarece conforme se profundiza en su estudio. La presente investigación se ha llevado a cabo realizando un análisis de las interacciones entre usuarios para su visualización con *Gephi* y posterior obtención de estadísticos.

Para lograr esta misiva, primeramente, es preciso encontrar una base de datos de una red social -Twitter en este caso-, para proceder a su descarga, tratado y visualización. Para este proyecto, el *dataset* seleccionado recibe el nombre de 'text_emotions' y contiene diferentes informaciones de 40.000 tweets.

Tras extraer la base de datos en formato CSV, es necesario procesar el archivo para facilitar su incorporación en *Gephi*, creando por ejemplo el archivo que contiene las aristas y atributos para las visualizaciones. Para lograrlo, se utiliza *Python* dando lugar a diferentes CSV, para poder crear las aristas y nodos, además de otros atributos relevantes. En última instancia, estos archivos son importados en *Gephi* para su visualización a base de ensayo y error con diferentes '*layouts'* para su posterior análisis de comunidades junto a la obtención de estadísticos para comprender con mayor precisión las posibles agrupaciones, comportamientos e '*insights'* de la población. Cabe destacar que los

métodos de importación y la selección de ‘layouts’ son cruciales para la construcción de visualizaciones.

A continuación, se procede a realizar una descripción más exhaustiva de los pasos metodológicos llevados a cabo para conseguir una disposición correcta de la información en *Gephi.*

3.1 Obtención del dataset y descripción de variables

Como se ha mencionado anteriormente, el objetivo principal se sitúa en torno al análisis de un dataset extraído de Twitter, para ello, se ha accedido a un repositorio de datos de uso público online denominado ‘Data.World’.

El *dataset* utilizado en este trabajo recibe el nombre de ‘text_emotion’, (crowdflower/sentiment-analysis-in-text) (1), se encuentra en formato CSV y recoge información de 40.000 tweets con el sentimiento que transmiten, pudiendo este tomar 13 valores diferentes. El modelo se obtuvo a través de un entrenamiento de *Deep learning*, incorporándose el dataset resultante como subset de datos para *Microsoft´s Cortana Intelligence Gallery (07-2016)* por *CrowdFlow* para uso de investigación. A continuación, se describen las variables que conforman el ‘text_emotions’:

-Tweet id: *número entero de 10 dígitos que identifica cada tweet, cada id es único para cada tweet.*
-Sentiment: *variable que determina el tipo de sentimiento que quiere transmitir un tweet. Su obtención ha sido a través de un modelo de ‘deep learning’. Esta variable categórica puede tomar 13 valores disintos (anger, boredom, empty, enthusiasm, fun, happiness, hate, love, neutral, relief, sadness, surprise, worry).*
-Author: *nombre de usuario que escribe un tweet en la red, cada tweet está identificado por su ‘tweet id’. Un mismo usuario puede tener diferentes tweets publicados, con diferentes menciones.*
-Content: *contenido del tweet que ha publicado cada ‘author’, esta cadena de texto recoge el mensaje de la publicación de cada tweet. Esta variable será utilizada para identificar las menciones de autores a otros posibles nodos, más adelante, se detallan más estas prácticas.*
Partiendo de la premisa de que las etiquetas de datos (‘sentiment’) se han obtenido con una metodología adecuada, es necesario, preparar el archivo con ciertas modificaciones que serán descritas en el siguiente apartado.

4. Tratado previo de datos

Es necesario proceder a un tratamiento previo de los datos, para ello, se ha escogido Python como herramienta, debido a su versatilidad y manejo para el trabajo con archivos en formato de texto. El código completo del archivo de Python se proporciona en el Anexo de este documento y será comentado a continuación para su total comprensión y conocimiento.

4.1 Limpieza preliminar en Python

Preview y limpieza: para confirmar que los datos se han importado correctamente, se muestran las primeras 4 filas del archivo *‘text_emotion’ (Fig. 1)*, además, se eliminan valores nulos si los hubiese - en este caso, los datos ya han sido tratados.
Figura 1. *Preview y limpieza de los datos.*

```python
import pandas as pd

# Cargar el archivo CSV
df = pd.read_csv('text_emotion.csv')

# Mostrar las primeras filas del dataset
print(df.head())
print()

# Verificar si hay valores nulos
print('Tabla de Valores nulos')
print(df.isnull().sum())
print()

# Eliminar filas con valores nulos si existen
df = df.dropna()
```

Figura 2. *Output consola.*

```
     tweet_id   sentiment       author  \
0  1956967341       empty   xoshayzers
1  1956967666     sadness    wannamama
2  1956967696     sadness    coolfunky
3  1956967789  enthusiasm  czareaquino
4  1956968416     neutral    xkilljoyx

                                              content
0  @tiffanylue i know  i was listenin to bad habi...
1  Layin n bed with a headache  ughhhh...waitin o...
2                Funeral ceremony...gloomy friday...
3               wants to hang out with friends SOON!
4  @dannycastillo We want to trade with someone w...
```

Fuente: *Elaboración propia*

En este punto, se podrían transformar a minúsculas los datos, pero este paso se realiza más adelante para reducir la carga computacional

4.2 Creación de nodos y aristas

Los nodos son todos los usuarios que integran la red, tanto los autores, cómo aquellos usuarios mencionados en sus tweets. Para su correcta extracción, se procede a crear una nueva columna llamada ‘Mentions’ que contiene el nombre de usuario mencionado en el tweet y, en su defecto, un espacio en blanco.

El nombre de usuario se encuentra localizando en cualquier posición de la columna ‘content’ símbolos que puedan representar una mención y son guardados en minúscula para evitar conflictos en el formato o duplicados.

También es necesario realizar una reducción de variables para facilitar la partición de datos en Gephi, aunque esta cuestión se explica con mayor detalle en el análisis explotarorio. En cualquier caso, se almacena el *data frame* modificado en formato CSV con el nombre de ‘aristas_autor_mencionados’ *(Fig.2).*

Figura 3. *Creación de nueva columna.*

```
# Crear una nueva columna con los valores reducidos
tweets_df['reduced_sentiment'] = tweets_df['sentiment'].map(mapping)

# Convertir 'Source' y 'Target' a minúsculas
tweets_df['author'] = tweets_df['author'].str.lower()
tweets_df['usuarios_mencionados'] = tweets_df['content'].apply(lambda texto: [user.lower() for user in re.findall(r'@(\w+)', texto)])
```

Fuente: *Elaboración propia.*

Este código, una vez ejecutado, se encarga de crear una nueva columna llamada 'reducted sentiment' en base a unas normas que serán descritas más adelante, además, convierte a minúscula el nombre de usuario de los autores y de las personas definidas como 'Target' (extraídas buscando una cadena de texto en las menciones *"@"*).

Por otra parte, es necesario obtener un listado de los usuarios que son autores y aquellos mencionados para crear 'nodes' -también en formato CSV para garantizar su compatibilidad con *Gephi* - que contiene un listado denominado ID, recogiendo los nombres de usuario de estos grupos *(Fig 3).*

Figura 4. *Creación de nodos y aristas.*

```
# Crear dataframe para las aristas (relación author -> usuarios mencionados)
aristas_autor_mencionados = tweets_df[['author', 'usuarios_mencionados', 'reduced_sentiment']].explode('usuarios_mencionados').dropna().reset_index(drop=True)

# Renombrar columnas para ser compatibles con Gephi
aristas_tweet_autor.rename(columns={'tweet_id': 'Source', 'author': 'Target'}, inplace=True)
aristas_autor_mencionados.rename(columns={'author': 'Source', 'usuarios_mencionados': 'Target'}, inplace=True)

# Guardar el listado de aristas en nuevos archivos CSV
aristas_tweet_autor.to_csv('aristas_tweet_autor.csv', index=False)
aristas_autor_mencionados.to_csv('aristas_autor_mencionados.csv', index=False)

# Visualizar el listado de aristas final
print('Aristas Tweet-Autor:')
print(aristas_tweet_autor.head())
print('\nAristas Autor-Mencionados:')
print(aristas_autor_mencionados.head())
```

Fuente: *Elaboración propia.*

Este código almacena por una parte las 'aristas_tweet_mencionadas.csv' y las "aristas_tweet_autor" que serán propuestas para un estudio futuro de coocurrencias. En este trabajo únicamente se incorporará a *Gephi* el archivo con las aristas que relaciona a los autores con los mencionados a través de un sentimiento.

4.3 Elaboración archivo 'target_counts'

En este punto, los archivos fundamentales para el análisis en *Gephi* están preparados. Para obtener *'insights'* más relevantes y poder aportar algún tipo de contexto a la red, se construye el archivo 'target_counts'. Este último archivo contiene un recuento y listado de todos los mencionados *(Fig. 5)*, así como el número de veces que es mencionado y el sentimiento que predomina en torno a sus menciones.

Con este archivo, se pretende proponer una forma automatizada de comprobar si existe relación de influencia (por número de seguidores) por parte de los mencionados, esta propuesta se detalla más adelante en 'Recomendaciones para futuros estudios'.
Además, es un documento útil para estudiar partes específicas de la población o llegar a conclusiones sobre comportamientos generales de la población, es posible incluso, determinar qué tipo de perfil se está visualizando en *Gephi* de acuerdo con el tipo de interacciones que se prevén.

Figura 5. *Crear 'target_counts'.*

```python
# Acceder al CSV 'aristas_autor_mencionados.csv' y contar las ocurrencias de cada valor en la columna 'Target'
edges_data = pd.read_csv('aristas_autor_mencionados.csv')

# Calcular el sentimiento predominante para cada Target
sentiment_mode = edges_data.groupby('Target')['reduced_sentiment'].agg(lambda x: x.mode()[0]).reset_index()
sentiment_mode.columns = ['Target', 'Predominant_Sentiment']

# Contar las menciones de cada Target
target_counts = edges_data['Target'].value_counts().reset_index()
target_counts.columns = ['Target', 'Count']

# Unir los conteos con el sentimiento predominante
target_counts = target_counts.merge(sentiment_mode, on='Target', how='left')

# Guardar los resultados en un nuevo archivo CSV
target_counts.to_csv('target_counts.csv', index=False)
print('\nConteo de menciones con sentimiento predominante:')
print(target_counts.head())
print()
```

Fuente: *Elaboración propia.*

En resumen, existen 3 CSV que serán manejados para ejecutar este trabajo, estos serán introducidos en el formato adecuado para su visualización con G*ephi*, para posteriormente, sacar conclusiones en base a la información extraída.

Text_emotion.csv: archivo que contiene el *dataset* original, es fundamental su correcto tratamiento porque condicionará el resto del proyecto. Aludiendo a lo mencionado anteriormente, este archivo sufre una modificación, añadiéndose la columna 'Mentions' para recoger la lista de usuarios que son mencionados y conformar los nodos de la red.
Aristas_autor_mencionados.csv: este archivo recoge la información dispuesta de forma que *Gephi* la reconoce y asigna los nodos y aristas de forma automática, para ello, se añaden las columnas 'Source' y 'Target', impuestas por el propio lenguaje de la herramienta para que la importación de aristas sea adecuada.

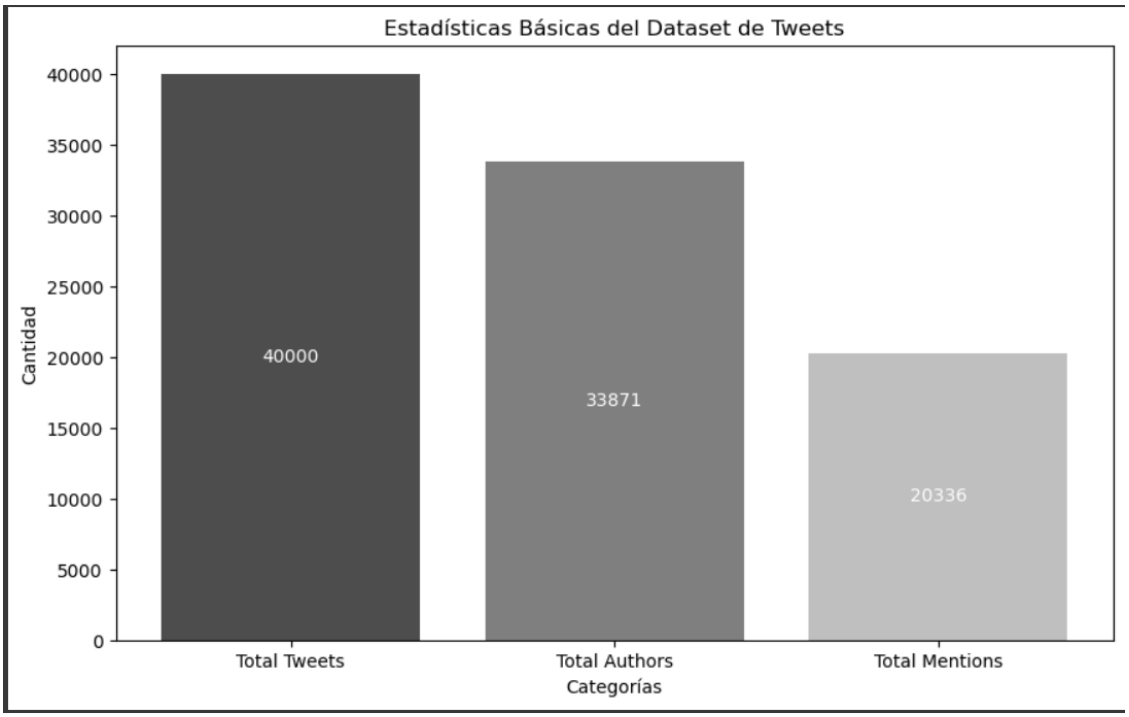
Target_counts.csv: contiene tres variables, una lista con todas las menciones realizadas, una cuenta del número de veces que se ha realizado esa mención y sentimiento (con la reducción realizada) que más predomina en sus conexiones.

5. Análisis Exploratorio de los datos

Con ánimo de obtener una imagen previa de las características y distribución de los datos de *'text_emotions'* se realiza un análisis exploratorio de las variables con estadísticas descriptivas básicas como el número de usuarios y de menciones presentes en la muestra. De esta forma, es más sencillo dirigir el estudio, entendiendo mejor cómo se comporta la muestra. De hecho, gracias a este estudio preliminar, se procede a reducir el número de etiquetas de sentimientos para que las visualizaciones en Gephi fueran más comprensibles. Adicionalmente, se examina la distribución de los tipos de sentimientos conociendo los atributos expresados en cada tweet para conocer cuáles son aquellos que predominan y poder establecer unas bases sólidas para la investigación.

5.1 Distribución de los nodos

Para comprender mejor la interacción entre los nodos, puede resultar útil conocer el número total de observaciones *('tweets') y* el número de valores único para 'author' junto al número de valores único para las menciones *(Fig. 6)*.
Figura 6. *Visualización estadística descriptiva.*



Fuente: *Elaboración propia.*

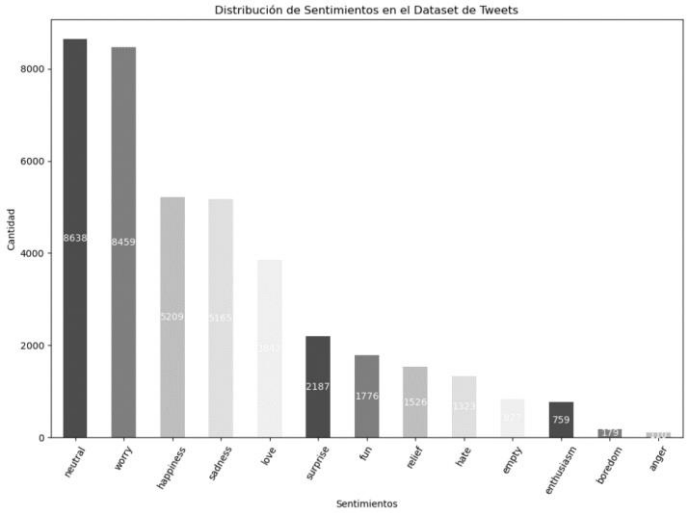
Como bien se ha comentado anteriormente, el número total de tweets es 40.000, aunque esto no significa que sea el número de nodos ya que un usuario puede escribir tweets que no puedan establecer otras relaciones con la muestra.

Por otra parte, la cantidad total de autores es de 33.871 y de menciones de 20.336, al suponer que ciertos usuarios tendrán influencia sobre otros, resulta intuitivo imaginar que muchos tweets mencionarán a una misma persona, pero esto no será algo recíproco, es por eso por lo que en *Gephi*, se propone un tipo de grafo dirigido al realizar la importación de datos.

5.2. Distribución de los sentimientos

En este apartado, se muestra la distribución que siguen los sentimientos en la muestra objeto de estudio *(Fig. 7)*. Se puede apreciar que los sentimientos más dominantes son 'neutral' y *'worry'* seguidos de *'happiness' y 'sadness'*. Estos valores serán tenidos en cuenta a la hora de analizar los nodos ('author-mention') ya que los 'inputs' que reciben determinan qué tipo de interacción tiene otros individuos hacia ese usuario, de esta forma, se puede contrastar qué tipo de perfiles generan un sentimiento positivo, negativo o neutral.

Figura 7. *Distribución de sentimientos.*



Fuente: *Elaboración propia.*

5.3 Reducción de sentimientos

Debido al elevado número de sentimientos, para facilitar la visualización de los datos en el grafo de *Gephi*, se procede a realizar una reducción de la columna 'sentiment', agrupándolos en 4 categorías: 'negative', 'positive', 'neutral' y 'surpirse'. Para ello se sigue la siguiente regla de sustitución:

Tabla 1. *Reducción de emociones.*

Término anterior	Término nuevo
anger	negative
boredom	neutral
enthusiasm	positive
fun	positive
happiness	positive
hate	negative
love	positive
neutral	neutral
relief	positive
sadness	negative
surprise	surprise
worry	negative

Fuente: *Elaboración propia.*

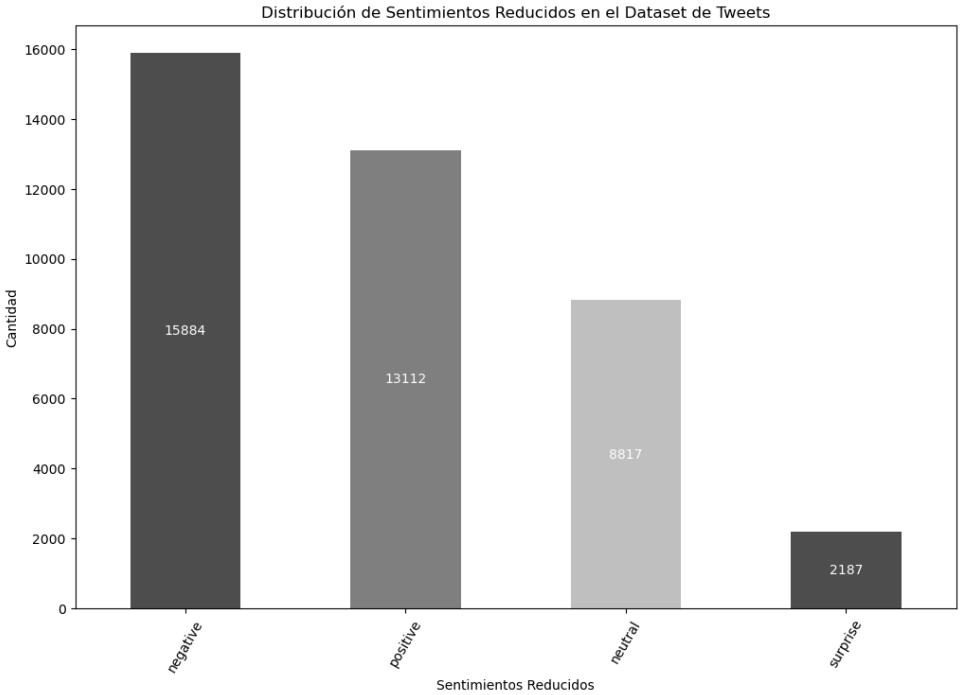
A continuación, se muestra el código para ello en Python, se procede a realizar un mapeo de estos sentimientos, después esta nueva variable se guarda en un *'dataframe'* que es convertido a minúsculas *(Fig. 8).* Este cambio, no afecta en ningún momento a la hora de establecer los nodos, simplemente, es un recurso para que la visualización en *Gephi* sea más sencilla. A continuación, se presenta la distribución de los cuatro sentimientos generados con este mapeo (*Fig. 9).*

Figura 8. Código de Python para reducir sentimientos.

```python
# Definir el mapeo de 13 a 5 categorías
mapping = {
    'anger': 'negative',
    'boredom': 'neutral',
    'empty': 'negative',
    'enthusiasm': 'positive',
    'fun': 'positive',
    'happiness': 'positive',
    'hate': 'negative',
    'love': 'positive',
    'neutral': 'neutral',
    'relief': 'positive',
    'sadness': 'negative',
    'surprise': 'surprise',
    'worry': 'negative'
}

# Crear una nueva columna con los valores reducidos
tweets_df['reduced_sentiment'] = tweets_df['sentiment'].map(mapping)
```

Fuente: *Elaboración propia.*

Figura 9. *Distribución de sentimientos reducidos.*



Fuente: *Elaboración propia.*

Tras este agrupamiento, se puede plantear la idea de que antes no fuese posible capturar los sentimientos reales de las personas o lo que buscaban transmitir debido a la cantidad tan elevada de etiquetas. Tras este nuevo cambio, la distribución de los datos cambia, resultando en:

-Negative 15.884 (35.09%)
-Positive 13.112 (34.63%)

-Neutral 8.817 (24.28%)
-Surprise 2.187 (6%)

Esta modificación en los datos servirá de gran ayuda para determinar qué tipo de relaciones y mensajes reciben principalmente los usuarios mencionados. En cualquier caso, a priori, parece una distribución de los datos con más sentido y cercana a la realidad: un balance entre opiniones positivas y negativas a la vez que un reducido tamaño de sentimientos neutrales o de sorpresa.

6. Visualización de datos en *Gephi*

Gephi impone dos condiciones para la importación de datos en la herramienta, la primera es que se encuentren en un archivo con un formato compatible - *GEXF, GDF, GML, GraphML, CSV entre otros…* -, la segunda condición dicta que los datos sean importados bajo la identificación de 'Source' y 'Target'.

En este caso, los datos han sido tratados de forma correcta previamente, resultando en un proceso de importación relativamente sencillo. Gracias a este manejo previo, es posible importar un archivo de aristas, de modo que la máquina automáticamente detecta todos los posibles nodos y completa la sección de *Gephi 'nodes'*.

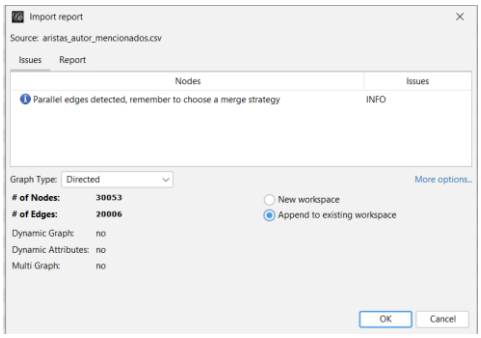
En las siguientes líneas se presenta paso a paso la metodología seguida para importar los datos y su correcta visualización.

6.1 Importación de datos

Atendiendo a las indicaciones del párrafo anterior, es preciso acceder a la pestaña 'edges' dentro del 'Workspace' en el que se está; es importante señalar que, por cuestiones de lenguaje, la primera y segunda columna de las aristas siempre han de recibir el nombre de 'Source' y 'Target' respectivamente, independientemente del origen de los datos, de modo que el programa pueda comprender que esa es la lista que comprende los nodos y sus atributos.

De esta manera, se carga el archivo CSV 'aristas_autor_mencionados.csv' que contiene una lista con todos los autores (Source) y el sentimiento (Attribute) que les relaciona con un mencionado (Target). Si estas variables se reflejan correctamente en la 'preview' de Gephi, como es el caso, al pulsar 'next' es necesario ajustar el tipo de grafo y el 'workspace' en el que se desea obtener la visualización *(Fig. 10).*

Figura 10. *Importación de aristas.*



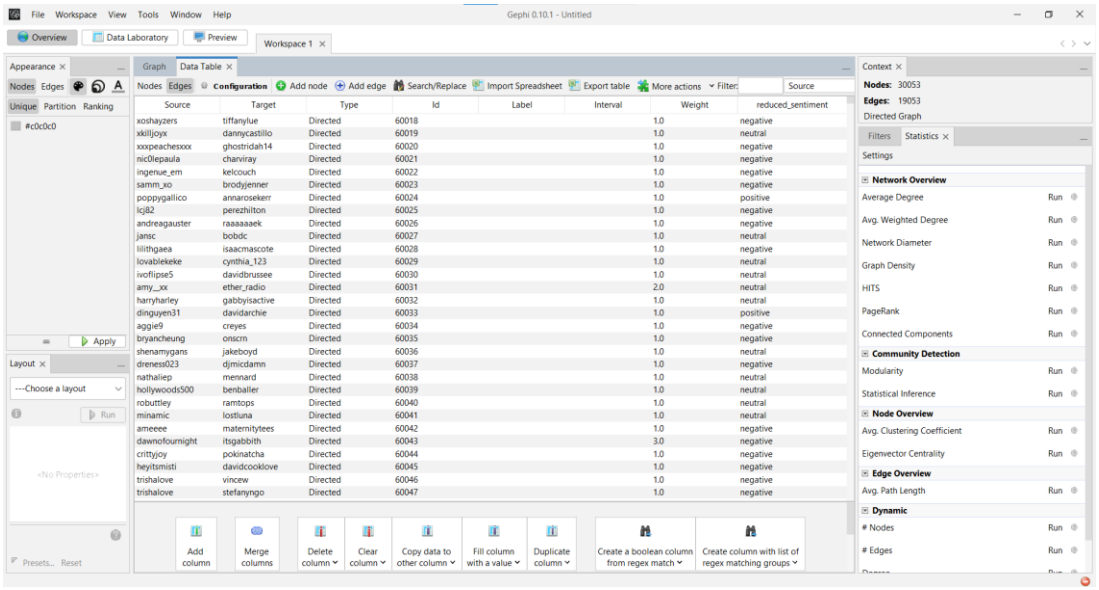
Fuente: *Elaboración propia.*

Para este *dataset*, las relaciones establecidas solo concurren en un sentido: los autores publican tweets con menciones a otras personas que existen en la red, pero los mencionados no tienen por qué responder a un autor, estos son los motivos que justifican la elección de tipo de grafo directo.

Por otra parte, gracias a esta pantalla emergente, es posible confirmar que la importación ha sido correcta en su totalidad. En este estudio, el número de nodos total es 30.053 (Personas que son autores de un tweet y aquellos que son mencionados), mientras que las aristas se componen por 20.006 relaciones en base al 'sentiment' que transmite cada *tweet (aristas).*

En la siguiente figura, se puede comprobar que la carga de datos ha sido correcta *(Fig. 11).* El siguiente paso consiste en preparar los nodos y aristas para su correcta visualización, además, hay que preparar cualquier atributo deseado y su etiquetado en la pestaña desplegable inferior 'labels'. En este estudio, estas etiquetas siempre serán ajustadas para facilitar la comprensión de la información.

Figura 11. *'Overview' datos en Gephi.*



Fuente: *Elaboración propia.*

6.2 Particiones por atributos

En este caso en particular, se ejecuta el 'Average Degree Report' para poder hacer una 'partition' de los datos de acuerdo con el *out-degree* (número medio de aristas por nodo) *(Fig. 12)*), para comprender mejor el número de conexiones medias que cabe esperar de cada nodo. En cualquier caso, ya se conoce que en esta distribución de población las aristas son muy dispersas.

A su vez, resulta interesante conocer la distribución de atributos frente a las aristas *(Fig. 13)*, de modo que para aquellos que tengan una cantidad representativa de nodos, se podrá determinar cuáles son los principales sentimientos que tiene la muestra hacia ellos, disponiendo de los sentimientos que mayormente predominan en su red/conexiones. Este tipo de 'insight' puede ser fundamental para identificar tendencias de algún tipo de sentimiento o aficiones comunes en la muestra.

Figura 12. *Nodes partition.*

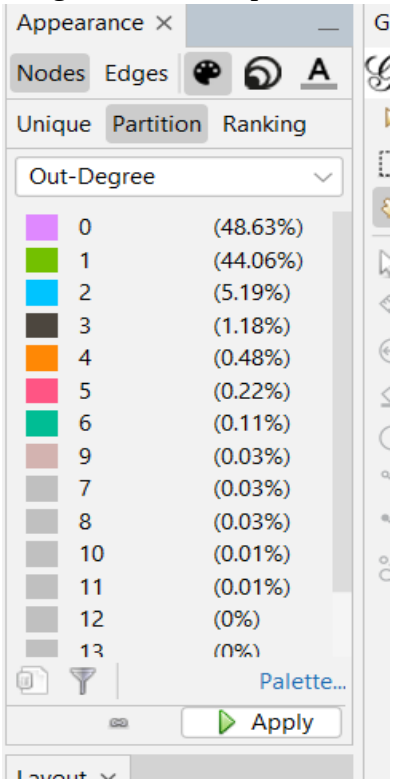
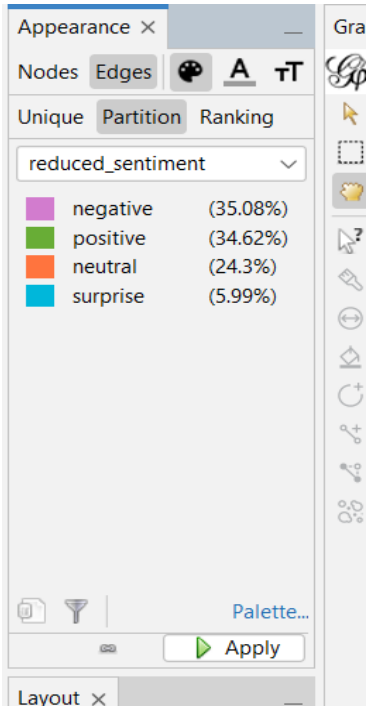


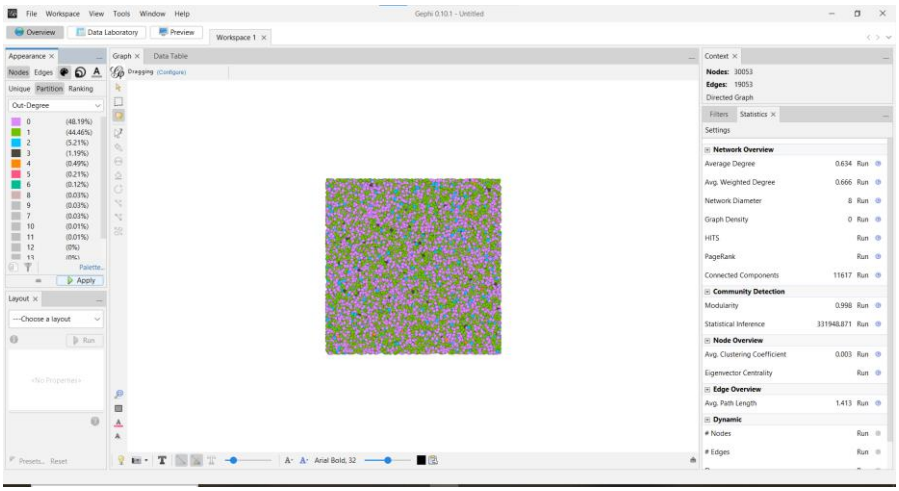
Figura 13. *Edges partition.*



Fuente: *Elaboración propia.*

Tras aplicar estas dos particiones de datos, es necesario proceder a correr diferentes *'layouts'* que aporten una visualización coherente, antes de ello, este es el aspecto que tiene el gráfico sin haber aplicado ningún *layout (Fig. 14)*

Figura 14. *Grafo con atributos.*



Fuente: *Elaboración propia.*

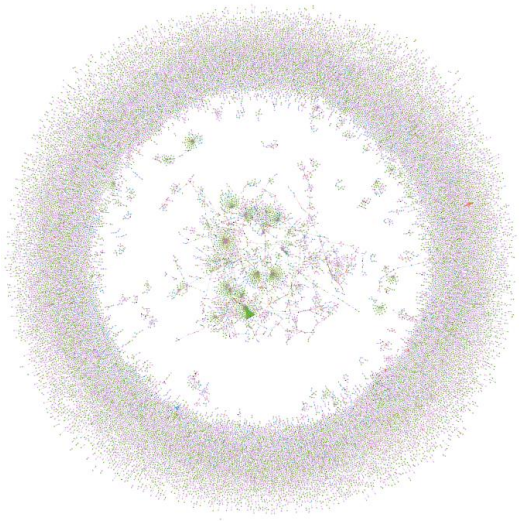
En este momento, se llega a la parte experimental del proyecto, en dónde, a base de ensayo y error, se ha de contrastar si la partición de datos aporta información de valor y elegir los 'layouts' aplicados para la visualización, estos detalles se exponen en los siguientes apartados con capturas de pantalla del proceso.

6.3 Aplicación de 'Layouts'

En el archivo de *Gephi* se pueden encontrar tres 'Workspace' diferentes, el primero corresponde a una visualización con 'Yifan Hu', el segundo, 'ForceAtlas2', mientras que el último, combina ambas disposiciones con ciertos parámetros ajustados.

Tras aplicar diferentes *layouts*, se concluye que la combinación que mejor ofrece una vista general de los datos resulta de *Yifan Hu* limitando el size a 30 y *ForceAtlas2* activando los parámetros 'LingLog mode' y 'Prevent Overlap' encontrado en el '*Workspace 3*' *(Fig. 15)*. Este es su resultado:

Figura 15. *Visualización inicial de la red.*



Fuente: *Elaboración propia.*

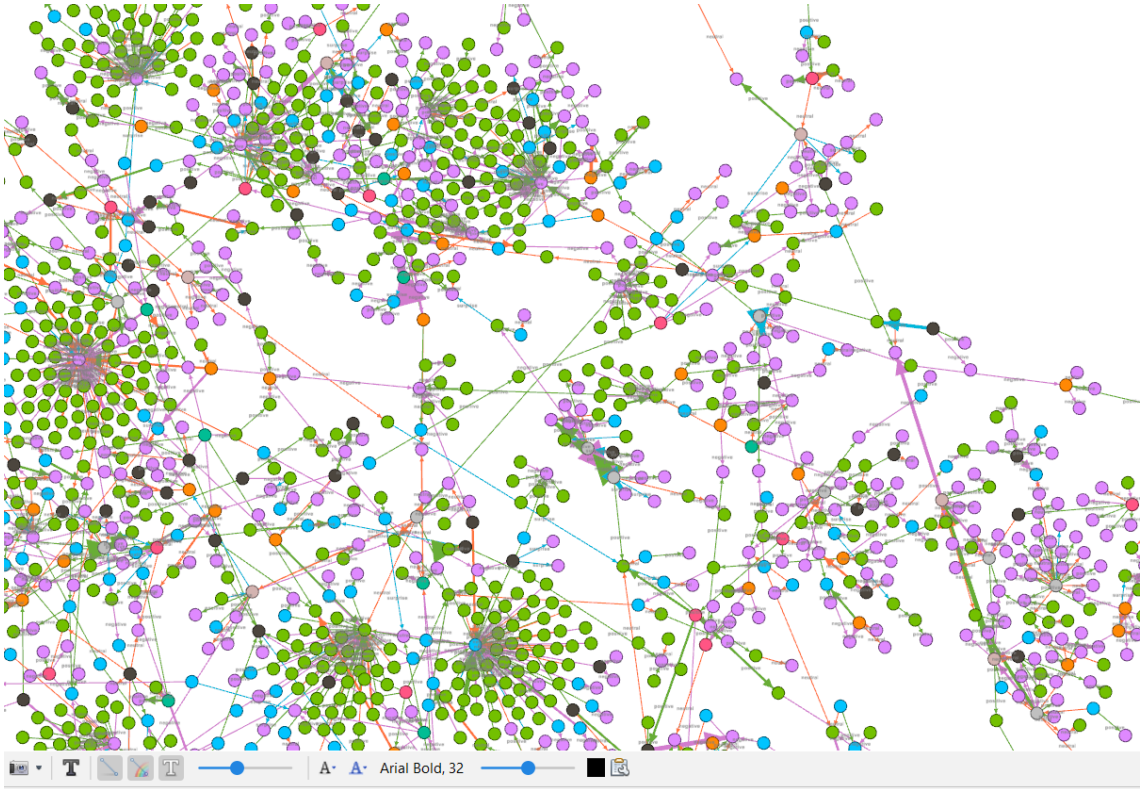
Con esta primera visualización, se pueden inferir varias cuestiones, en primer lugar, la mayor parte de la muestra tiene conexiones muy débiles y poco conectadas (anillo exterior), mientras que existen ciertos grupos en las zonas centrales, aunque sus nodos

son principalmente de una (*color verde*) o dos (*color azul*) conexiones, aunque algunos pueden llegar a tener más de 10 junto a sus atributos ('sentiment'; *Fig. 16).*

La segunda conclusión que se puede extraer es que la mayor parte de nodos tienen muy pocas relaciones con otros, lo que podría significar que existen pocas relaciones cercanas, aunque sí que existen atracciones de diferentes autores en torno a una misma fuente.
De la misma manera, existen otras dos visualizaciones para comprender mejor las relaciones entre los datos.

Por una parte, en el *'Workspace 1'* un 'layout' inicial realizada con 'Yihan fu', una de las más comunes para poder obtener una visualización inicial de la información. Mientras que, en el 'Workspace 2, se combinan las 'layouts' al igual que en el tercer apartado pero se limita la distancia de los edges. En el anexo de este trabajo, se facilitan capturas de pantalla con visualizaciones de estos dos 'Workspace'.

Figura 16. *Muestra de atributos y 'Workspace 3'.*



Fuente: *Elaboración propia.*

A continuación, se abunda en los detalles de ambas zonas, por una parte, el área exterior de menor conexión de la red, y por otra, en mayor profundidad, de la zona central, sus nodos y aristas, así, se podrá ofrecer en detalle el tipo de conexiones y sentimientos que se establecen en torno a una figura presumiblemente reconocible.

6.4 Estudio de comunidades

Atendiendo a lo comentado con anterioridad, el estudio de comunidades se divide en dos, por una parte, aquellos nodos que conforman el anillo exterior y que tienen un porcentaje

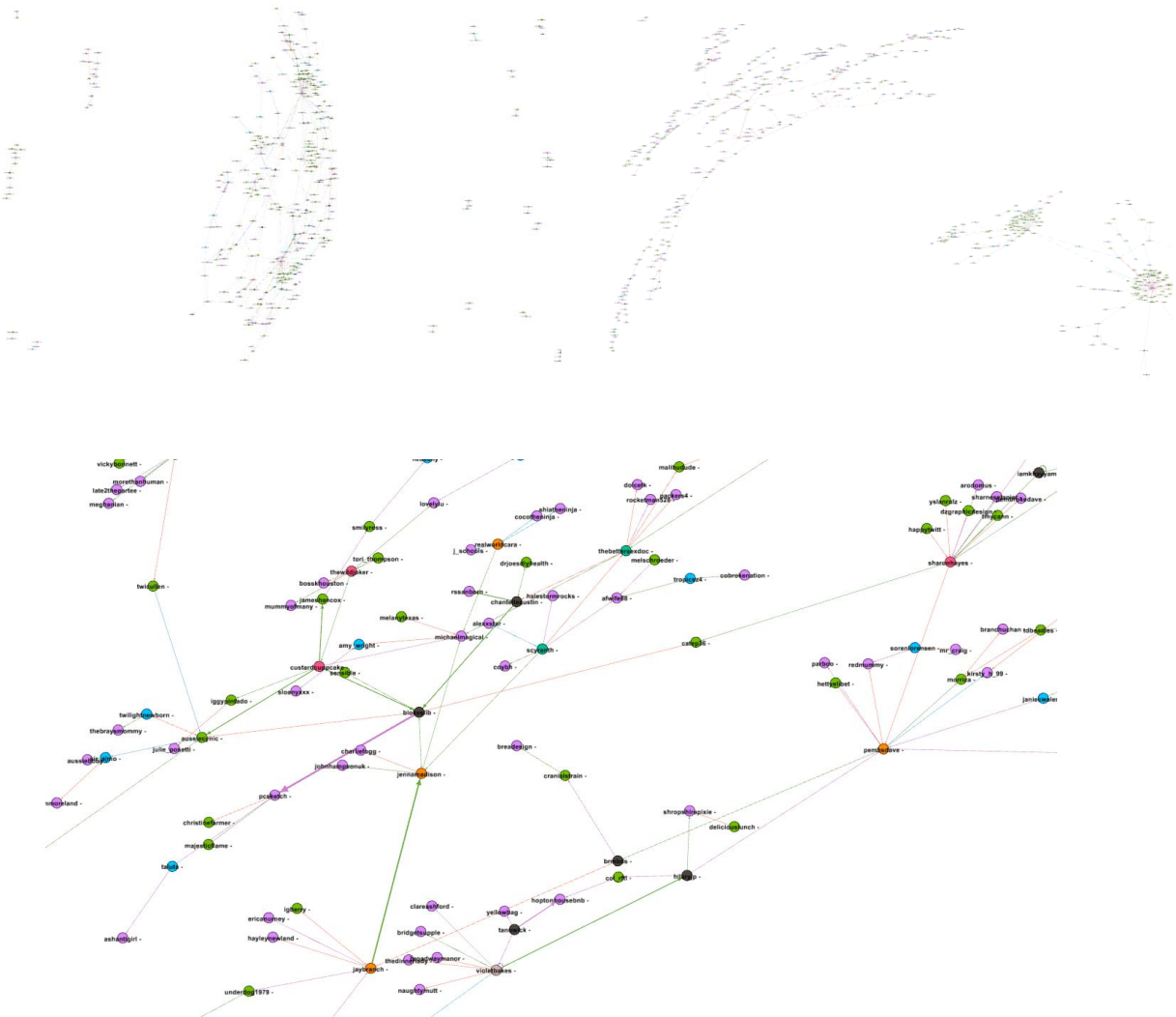
de interacción menor y, posteriormente, en exhaustivo detalle, las aristas que conforman las agrupaciones interiores.

6.4.1 Sector externo

Al indagar en esta sección, la información se encuentra disgregada e inconexa, generalmente no existe más de una relación ni construcciones de aristas complejas.

Parece haber ciertas tendencias y pequeños agrupamientos en torno a ideas y sentimientos entre autores y sentimientos. Para entender mejor cómo funcionan estas pequeñas agrupaciones, se aplica en un nuevo 'workspace' la layout 'ForceAtlas2' con los parámetros 'LingLog mode' y 'Prevent Overlap' activados *(Fig. 17, Fig. 18)*, la visualización resulta en agrupaciones pequeñas en las zonas exteriores con la siguiente forma:

Figura 17. *Agrupaciones exteriores con ForceAtlas2.*



Fuente: *Elaboración propia.*

Cabe destacar que, 3 usuarios que se encuentran entre los 10 más mencionados, construyen este tipo de agrupaciones en sectores más externos, aunque serán analizados posteriormente debido a su capacidad de influencia más adelante se encuentran en estas dos zonas más alejadas del centro, sus nombres de usuario son *@davidarchie, @jonathanrknight y @retrorewind.* Esto es un factor considerable a la hora de

contextualizar a la muestra, ya que podría indicar que su contexto, aunque esté relacionado, sea diferente al de los otros Targets con mayor agrupamiento.

En definitiva, las agrupaciones en la zona exterior son generalmente grupos muy pequeños unidos por un sentimiento común, el color de las aristas suele ser el mismo para diferentes nodos conectados. Existen interacciones entre autores y mencionados, pero no hay un tamaño significativo que merezca una mención mayor. Estos individuos resultan interesantes para un estudio enfocado a la conversión de esos usuarios que no están en consonancia con las agrupaciones más destacadas.

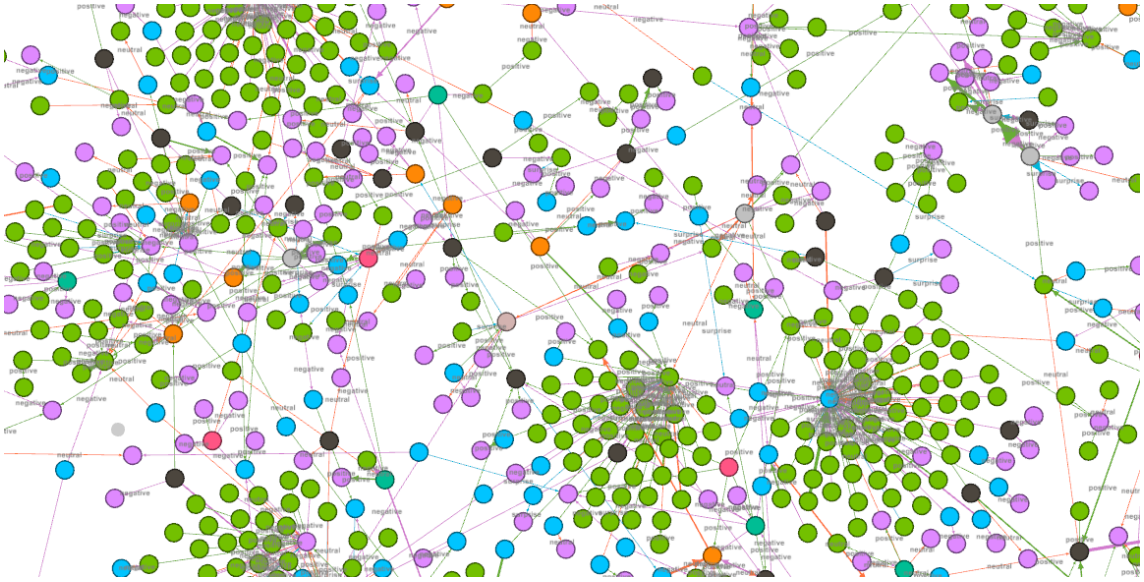
Los sentimientos que más predominan son los negativos y positivos, esto resulta inconcluyente, pero asienta las bases para plantear la hipótesis de que un porcentaje significativo de los tweets sean entre contactos cercanos (positivos), mientras que aquellos negativos pueden ser con usuarios desconocidos devengados de discusiones públicas.

6.4.2 Sector interno

La sección interna ofrece una mayor variedad en cuanto al número de nodos, tipo de relaciones y sentimientos *(Fig. 19)*. Los datos se agrupan en torno a menciones comunes, reportando todo tipo de sentimientos.

Si bien es cierto que la mayoría de las relaciones tienen una única conexión, se pueden encontrar nodos de más conexiones con cierta facilidad, esto implica que hay factores comunes a estos usuarios que de una forma u otra, hace que interactúen transmitiendo sus emociones.

Figura 19. *ForceAtlas2 - Sección interna.*



Fuente: *Elaboración propia.*

Para conocer los individuos que pueden tener cierta representatividad en los comentarios, se preparó el archivo *'target_counts.csv'* que recoge para cada mencionado, el número de menciones que recibe en orden descendiente y el sentimiento que más predomina en sus menciones *(Fig. 20)*.

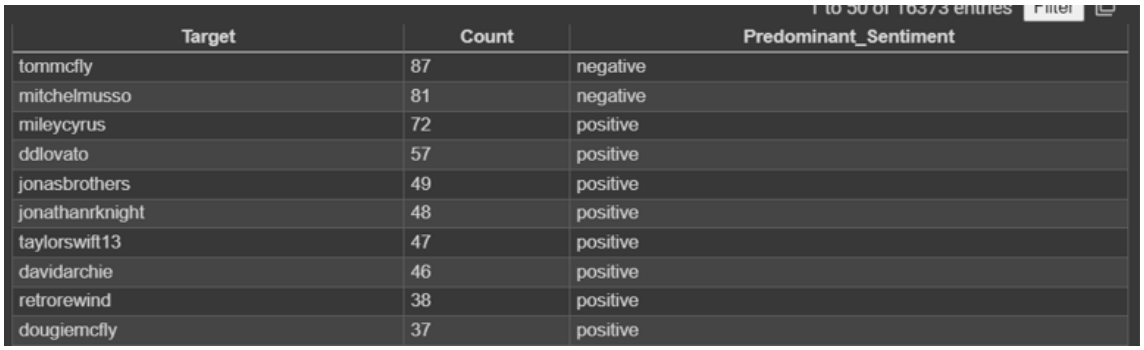
Figura 20. *'Head' de target_counts.csv.*

```
Conteo de menciones con sentimiento predominante:
          Target  Count Predominant_Sentiment
0       tommcfly     87              negative
1   mitchelmusso     81              negative
2     mileycyrus     72              positive
3       ddlovato     57              positive
4  jonasbrothers     49              positive
```

Fuente: *Elaboración propia.*

Uno de los objetivos del estudio consiste en detectar a aquellas personas de mayor influencia para analizar cuál es su interacción con la red. A continuación, se ofrece una vista detallada de los 10 nodos más relevantes según el número de veces que son mencionados.

Figura 21. *Top 10 más mencionados.*



Target	Count	Predominant_Sentiment
tommcfly	87	negative
mitchelmusso	81	negative
mileycyrus	72	positive
ddlovato	57	positive
jonasbrothers	49	positive
jonathanrknight	48	positive
taylorswift13	47	positive
davidarchie	46	positive
retrorewind	38	positive
dougiemcfly	37	positive

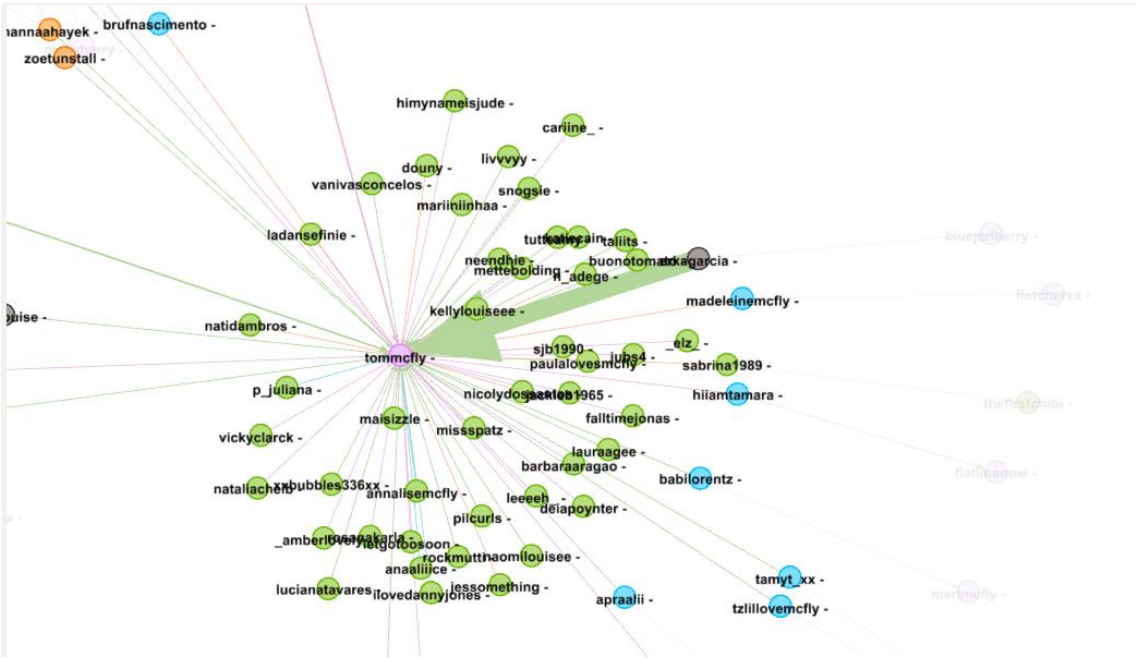
Fuente: *Elaboración propia.*

Al igual que el resto de los documentos adicionales al proyecto, el código de Python para obtener este CSV se puede encontrar en el Anexo del trabajo.

Una vez obtenida una lista con los usuarios que más menciones reciben, es fácil contrastar qué estos usuarios tienen un número significativo de aristas conectándoles con otros nodos. De hecho, sin necesidad de esta lista, es posible encontrar fácilmente a estos 10 ‘Major Players’.

A lo largo de los siguientes párrafos, se seguirá una estructura iterativa, presentando una captura de pantalla para cada uno de estos usuarios y un comentario tras esta imagen con las posibles conclusiones que se pueden extraer de cada red de interacción.

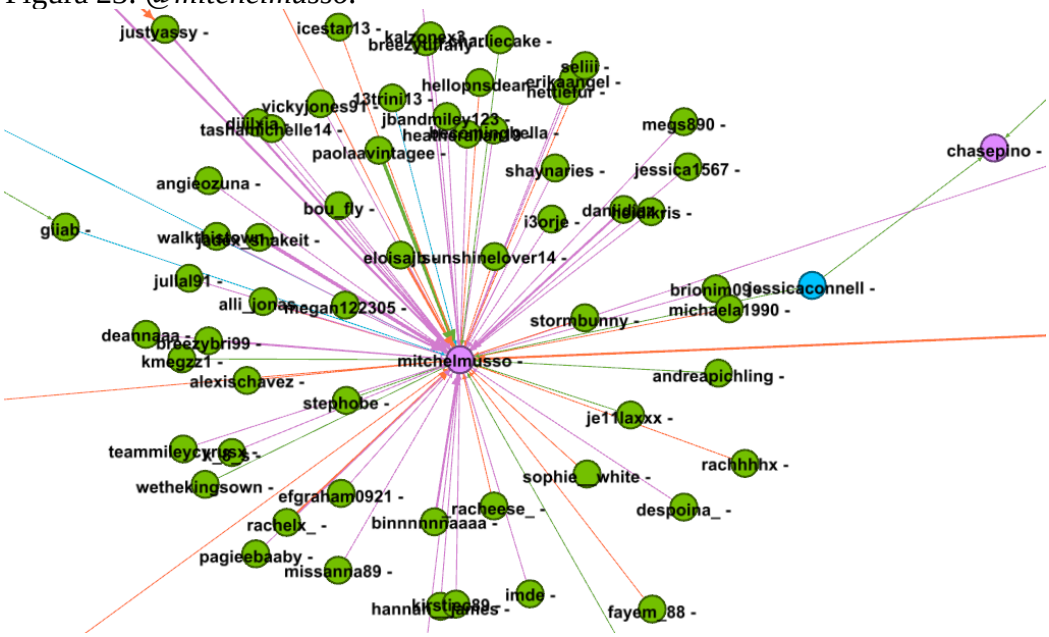
Figura 22: *@tommcfly.*



Fuente: *Elaboración propia.*

Este usuario es el más mencionado de la red, a pesar de ello, eso no implica que sea el más querido, se puede apreciar (*en rosa*) que el sentimiento que predomina la mayor parte de nodos que le conectan tiene un *'out-degree'* igual a 1, es decir, son individuos que solo establecen una relación, en este caso con el usuario *@tommcfly*. El hecho de que no existan relaciones entre los nombres de usuarios y el sentimiento que predomina en las aristas, puede ayudar a plantear la idea de que este usuario haya publicado algún tipo de contenido ofensivo, recibiendo así una ola de respuestas negativas.

Figura 23: *@mitchelmusso.*

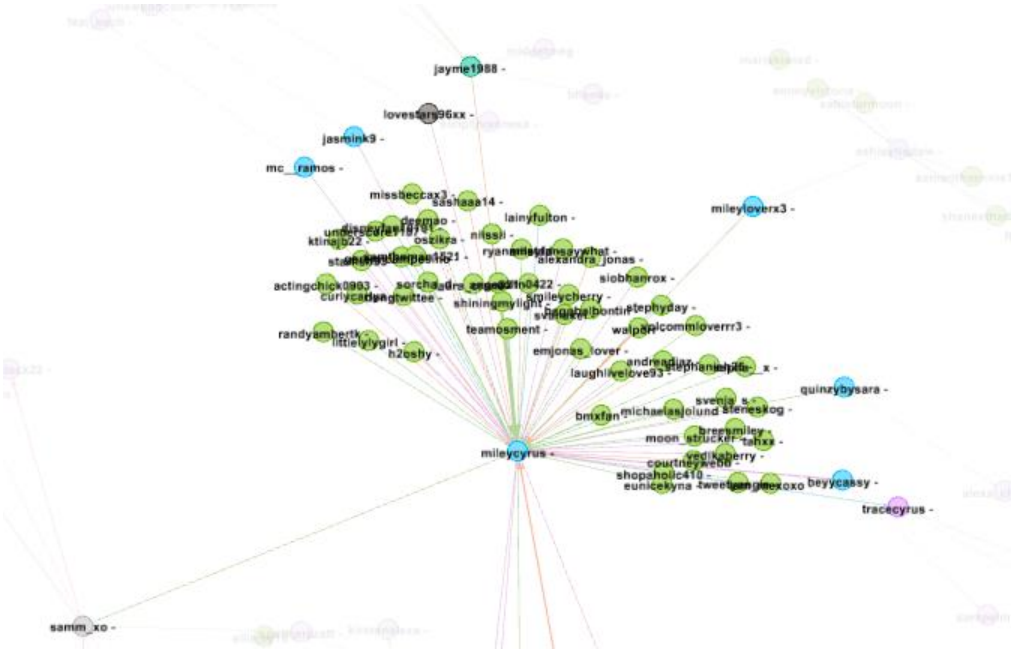


Fuente: *Elaboración propia.*

Este caso es similar al anterior, reinan con cierta intensidad los atributos negativos, de la misma manera, se puede apreciar la presencia de una cantidad significativa de nodos con

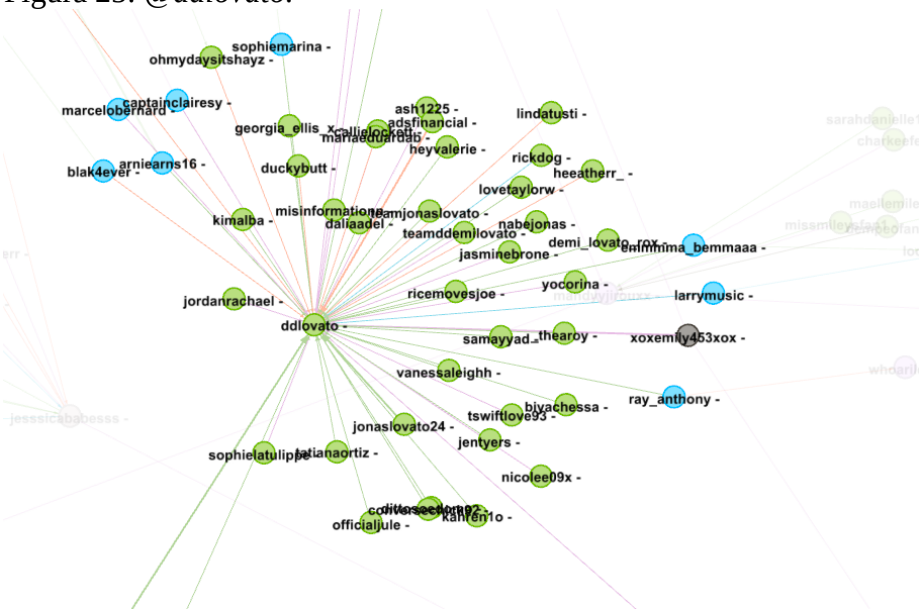
un sentimiento neutral. Este tipo de atributo podría devengar de ciertos usuarios tratando de defender al mencionado o simplemente divulgando algún tipo de información relacionada con esta persona.

Figura 24: *@mileycyrus.*



Este es el primer usuario en conseguir que el sentimiento promedio transmitido sea positivo, sus nodos son principalmente verdes aunque solo establecen una conexión, esto podría sugerir que siguen a @mileycyrus y es lo único que tienen en común.

Figura 25: *@ddlovato.*

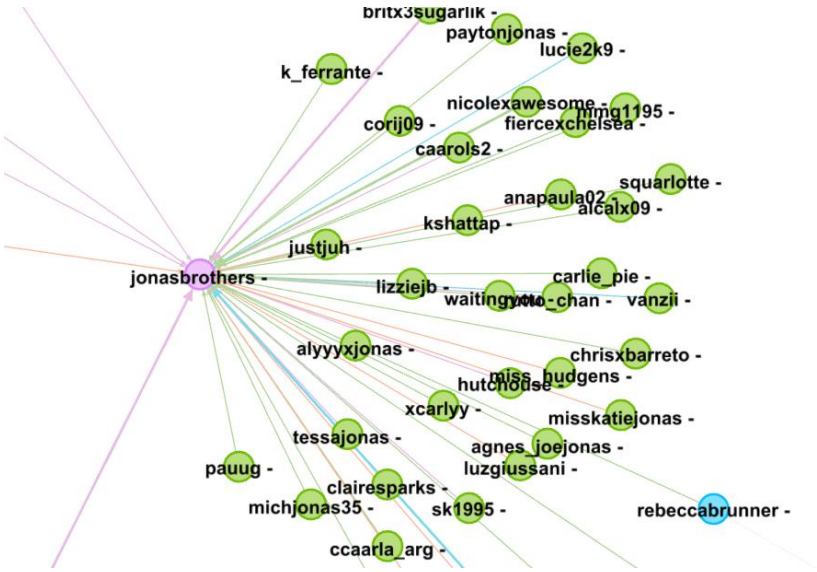


Fuente: *Elaboración propia.*

@ddlovato es el primer nodo que demuestra una relación entre los mencionados dentro de un contexto de influencia ya que existen nodos que utilizan el nombre del target alterado como nombre de usuario (*'@demi:lovato_rox-)*, de hecho, algunos de los nodos 'Source' contienen nombres de usuario con variaciones de otros mencionados

('@tswiftlove93'), este hecho, además de influencia en la red por parte de los mencionados, puede sugerir una relación del contexto por el que estos 'Target' son mencionados.

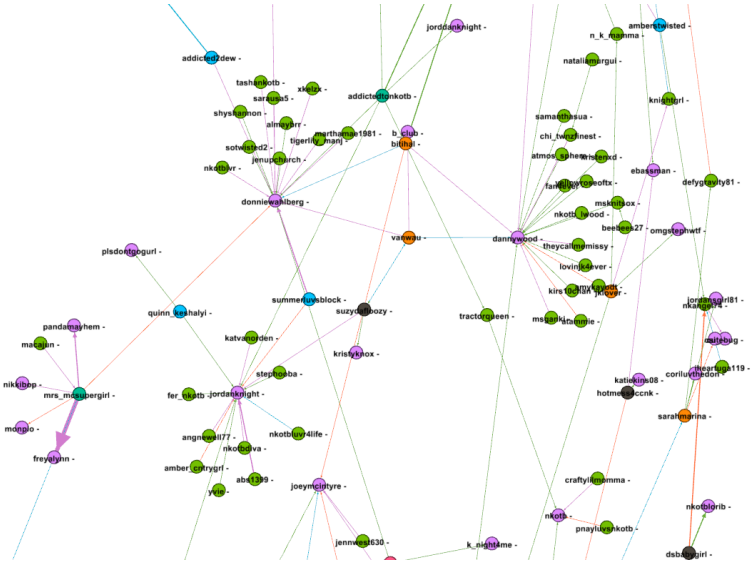
Figura 26: *@jonasbrothers.*



Fuente: *Elaboración propia.*

A medida que se indaga en los datos, se hace más evidente el hecho de que, los nodos que conforman los 'Targets' con mayores menciones son, generalmente, personajes públicos que son mencionados por sus seguidores o 'haters'. En el caso particular de *@jonasbrothers*, las aristas revelan diversos tipos de atributos, en cualquier caso, se cumple la tónica general de que todos aquellos que contienen la cadena 'jonas' como parte de su nombre de usuario reportan sentimientos principalmente positivos.

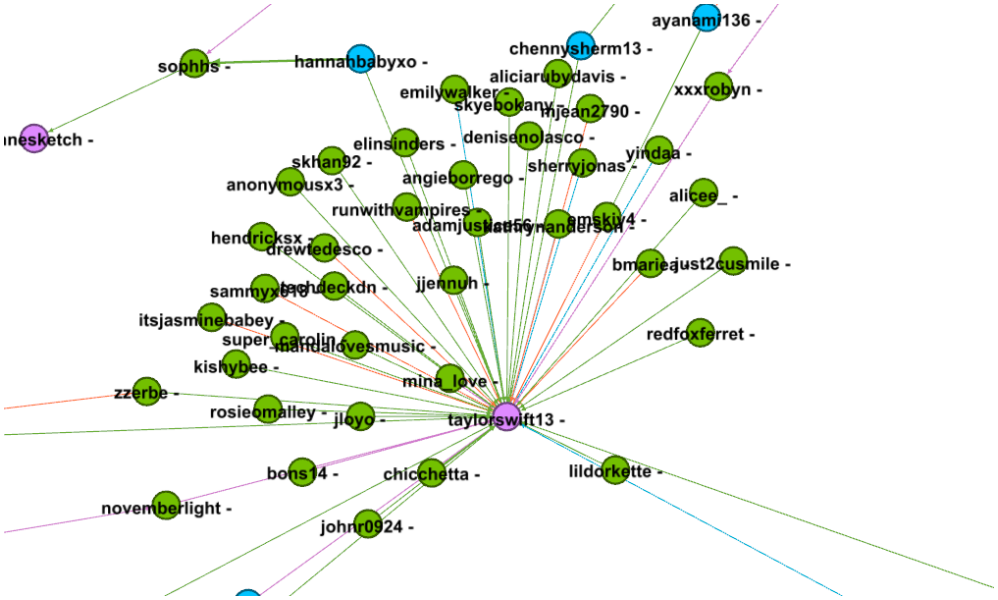
Figura 27. *@jonathanrknight.*



Fuente: *Elaboración propia.*

En este caso, la posición del usuario y la distribución de la red a su alrededor varía, todo tipo de sentimientos y número de enlaces, esto podría indicar que este autor se dedica a la publicación de opiniones o similar debido a la alta variedad en la distribución de los datos.

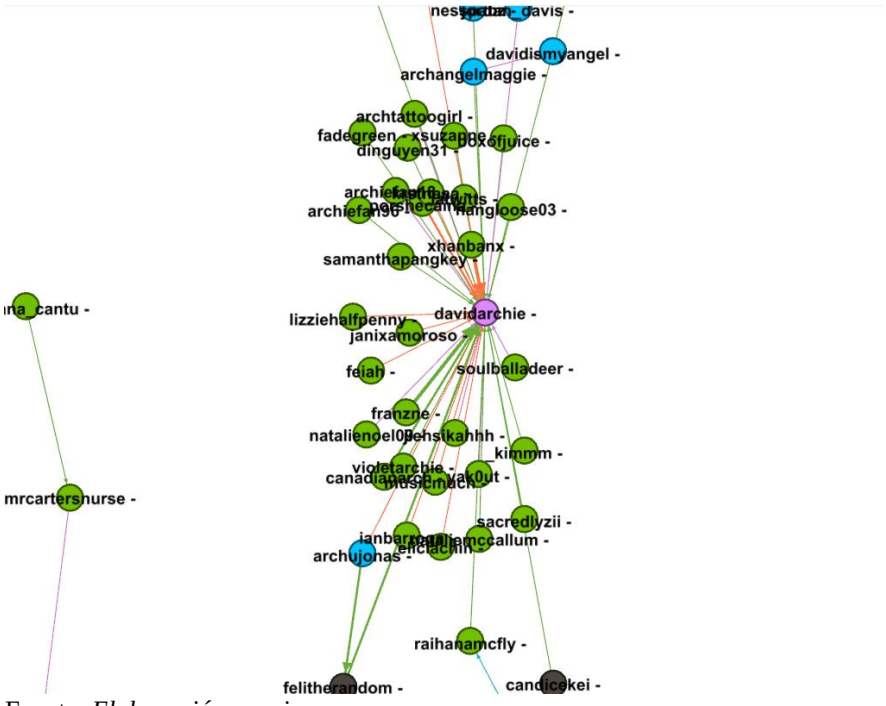
Figura 28. *@taylorswift13.*



Fuente: *Elaboración propia.*

De nuevo, resulta inferible que *@taylorswift13* podría ser un personaje público. La mayoría de aristas tienen una flecha verde, por lo que de nuevo, los sentimientos positivos son predominantes hacia esta figura.

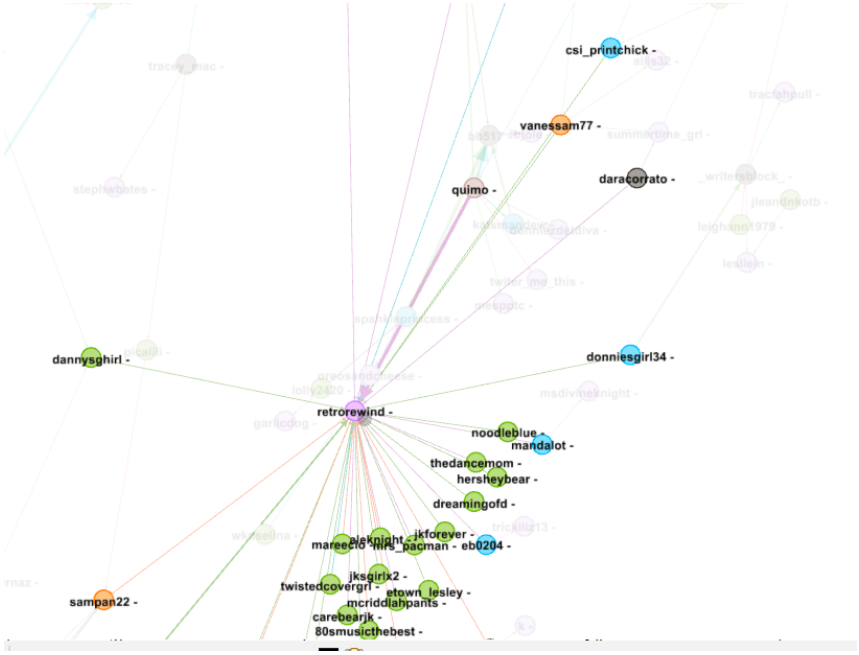
Figura 29: *@davidarchie.*



Fuente: *Elaboración propia.*

Este autor comparte características en su interacción con la red con el usuario *@jonathanrknight* sus nodos comparten varias conexiones y generalmente son sentimientos positivos.

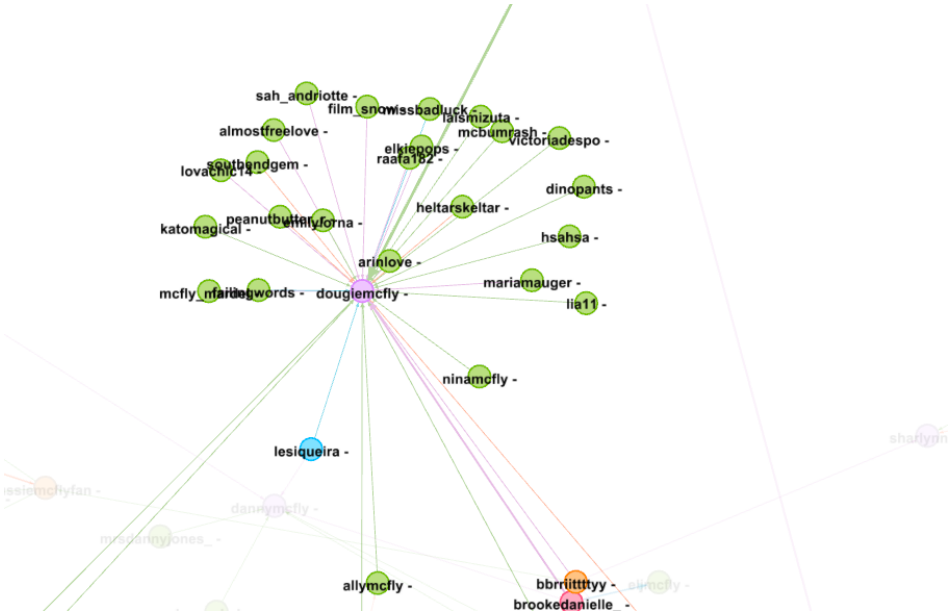
Figura 30: *@retrorewind.*



Fuente: *Elaboración propia.*

De nuevo se repiten características comunes que son indicativos del tipo de usuario de esta cuenta. Podría tratarse de un medio de divulgación de nicho debido a su *feedback*, si bien es cierto que existe una vertiente negativa, la mayoría de sus inputs son positivos o neutros.

Figura 31: *@dougiemccfly.*



Fuente: *Elaboración propia.*

Este último usuario recibe primordialmente un feedback positivo, de nuevo, aparecen nodos con nombres de usuario que contienen la cadena de texto de la arista central. Podría tratarse nuevamente de un personaje público.

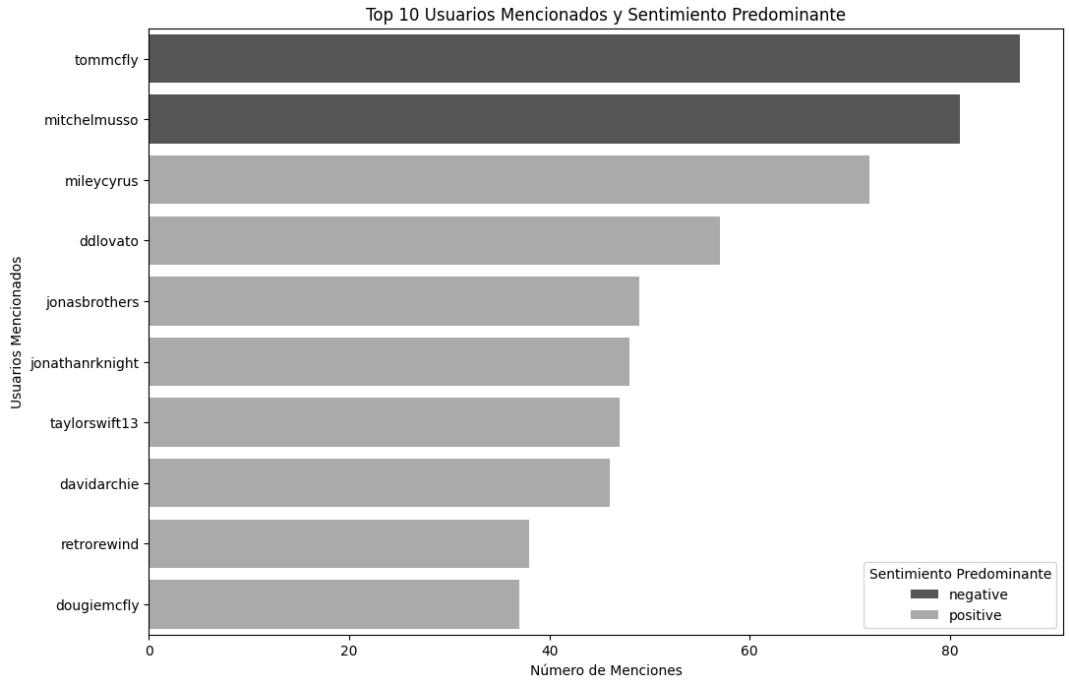
Para poder resumir toda esta información se procede a elaborar un código de python *(Fig. 32)* para generar un gráfico *(Fig. 33)* que contenga una lista con los 10 nombres de usuario que se han estudiado, la cantidad de menciones que tienen y las emociones que reciben por parte de los nodos.

Figura 32. *Código para generar gráfico Top 10.*

```
# Seleccionar los 10 usuarios con más menciones
top_10_users = target_counts.head(10)

# Crear el gráfico
plt.figure(figsize=(12, 8))
sns.barplot(data=top_10_users, x='Count', y='Target', hue='Predominant_Sentiment', palette='gray')

# Configurar etiquetas y título
plt.xlabel('Número de Menciones')
plt.ylabel('Usuarios Mencionados')
plt.title('Top 10 Usuarios Mencionados y Sentimiento Predominante')
plt.legend(title='Sentimiento Predominante')

# Mostrar el gráfico
plt.show()
```

Fuente: *Elaboración propia.*

Figura 33. *Gráfico Top 10 Usuarios con más menciones.*



Fuente: *Elaboración propia.*

Este gráfico muestra un resumen de la información presentada en Gephi. Si bien es cierto que la mayoría de Targets reciben como input principal sentimientos y opiniones positivas, cabe destacar la presencia de los negativos en los dos primeros usuarios. Este

gráfico muestra una predominancia de lo positivo, aún así, los dos valores más altos provienen de fuentes negativas. Estas cuestiones serán discutidas más adelante en el apartado de conclusiones.

7. Generación de estadísticos

Para ofrecer una mejor comprensión de los datos y la interacción entre los nodos, *Gephi* facilita la ejecución de estadísticos, estos, serán descritos brevemente en los siguientes apartados junto al resultado obtenido para la muestra tratada. Para facilitar la lectura de estos indicadores, su visualización gráfica se encuentra en el anexo y los archivos generados con la entrega de este documento en un enlace a SharePoint.

Average Degree – 0.634
Este estadístico muestra el número medio de aristas que tiene cada nodo, es decir el número medio de enlaces que tiene cada nodo de la red. Un valor tan bajo indica, como se ha indicado a lo largo de la discusión una media de enlaces y conexiones muy baja. En cualquier caso, eso no supone un impedimento para el estudio de los datos.
Por otra parte, es preciso comentar la segmentación indegree y outdegree de la población. En este caso en particular, se ha realizado una partición de los datos en función de la 'outdegree distribution' para identificar qué nodos actúan con cierta capacidad de influencia. Las gráficas para esta distribución pueden ser encontradas en el anexo así como el valor para la media ponderada (0.666) y su distribución.

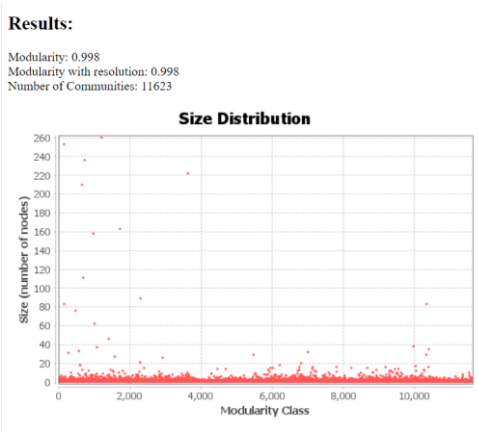
Clustering coefficient – 0.003
Este valor indica la probabilidad de que dos nodos sean vecinos. Para el dataset utilizado es bastante bajo, en futuros estudios, se podría mejorar en este aspecto si se desease asignando un peso diferente a las conexiones o forzando una mayor afinidad de los datos utilizando recursos como la API de Twitter para introducir nuevas relaciones y atributos, por ejemplo, tener en cuenta el número de seguidores o los intereses de la muestra.

Connected components – Weakly: 11.617 **| Strongly:** 29.668
Este estadístico muestra la capacidad para describir la conexión de los componentes en la red. En el caso de esta red, existen un número muy alto de componentes conectados, lo que supone que la fragmentación de datos sea elevada, dando lugar a muchos subgrupos que no están conectados entre sí. Si se introdujesen contactos comunes, sería otro nodo para poder conectar a la red de una forma más densa. Para esta red estudiada, la densidad de red es muy cercana a cero por estos factores.

Eigenvector Centrality Report - 0.743562355531949
Este estadístico trata de relevar la importancia de las conexiones, aquellos nodos con valores más cercanos a 1 representan una importancia mayor en la red mientras que los más cercanos a 0 contienen una centralidad de vector baja lo que significa una conectividad baja.

Modularity Report – 0.998
La modularidad es uno de los estadísticos de mayor relevancia ya que provee una vista de la estructura de la red atendiendo a las comunidades que la conforman, un valor tan alto de modularidad indica que la conectividad dentro de cada módulo es muy alta pero que existen pocos enlaces entre los módulos como se ha demostrado anteriormente. Por otra parte, esto inequívocamente hace que el número de comunidades sea tan alto. A continuación, se muestra el resultado del report de modularity *(Fig. 34).*

Figura 34. *Modularity Report.*

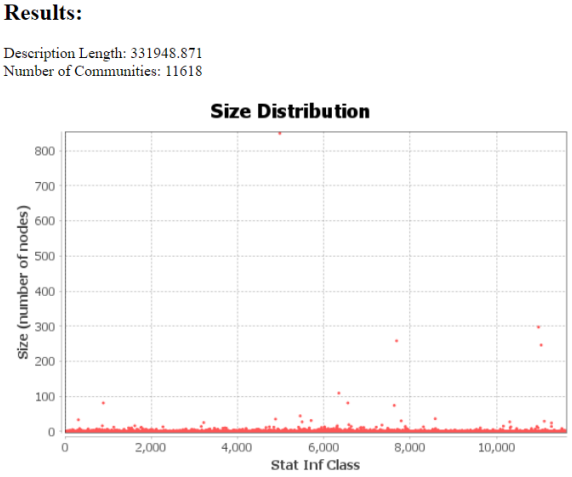


Fuente: *Elaboración propia.*

Graph Distance Report – 8 | APL 1.413093904975204
Teniendo en cuenta la morfología de la red, es de esperar que su diámetro sea amplio. Esto indica que la red tiene pocas conexiones, en este caso en particular, la distancia más larga ente dos nodos es de 8. Para reducir el diámetro de una red se pueden llevar a cabo diferentes estrategias, por ejemplo, establecer contactos intermedios que faciliten conexiones directas entre la red o establecer una jerarquía en las redes. Por otra parte, el '*Average Lenght Path'* es cercano a 1, lo que indica una alta cohesión entre la media de las distancias más cortas de los nodos.

Statistical Inference Report – 331.948,871
En este estadístico cabe destacar el elevado número de bits necesarios para describir la partición de una red. Si se hubiese seleccionado un *subset* de datos compuesto por las relaciones más conectados es posible reducir el valor del estadístico *(Fig. 35)*, a pesar de ello, en este trabajo se ha querido ofrecer una visión tanto de los nodos influyentes como de aquellos que no son tan significativos.

Figura 35. *Statistical Inference Report.*



Fuente: *Elaboración propia.*
Si se diese el caso de que el lector quiera conocer algún otro estadístico, puede consultarlo en el Anexo del trabajo, además, en el enlace al SharePoint - alojado en el Anexo -, puede

consultar y ejecutar tanto el código, cómo el archivo de *Gephi* con el que se han realizado estas visualizaciones.

8. Conclusiones

En este apartado final, se procede a la elaboración de conclusiones en referencia a este TFG de Business Analytics, incluyendo recomendaciones y consideraciones para futuros estudios en base a los detalles y experiencias aprendidas durante la elaboración del proyecto.

8.1 Recomendaciones para futuros estudios

A lo largo de este trabajo, surgen diferentes cuestiones que se pueden implementar en estudios siguientes; en primer lugar, una agrupación de la muestra o ampliación con nodos interconectados para facilitar la interacción entre la red y los atributos que unen a cada persona.

Por otra parte, la disposición del 'dataset' aristas_tweet_author, que está preparado para realizar un análisis de texto, relacionando las posibles coocurrencias que se den en lo mensajes y comprobando si existe relación entre ellas y los sentimientos. Se pretendía haber incluido este análisis en este trabajo, pero por cuestiones de logística y tiempo, ha sido algo inviable.

Otro aprendizaje para considerar es que, las particiones de datos determinan totalmente la apariencia de una visualización, es por ello, que no solo conviene matizar en los atributos extraídos de los textos, sino en el tipo de particiones que se realiza a los nodos.

Finalmente, otra cuestión que se pretendía incluir es la interacción con el complemento de *Gephi* para Twitter, para poder recabar datos a través de este 'plug-in', pero, las políticas de Twitter han cambiado y este tipo de información no es accesible de manera gratuita actualmente, este también es el motivo, por el que ha resultado de elevada complejidad el conseguir un set de datos que se adecuase a las necesidades del estudio.

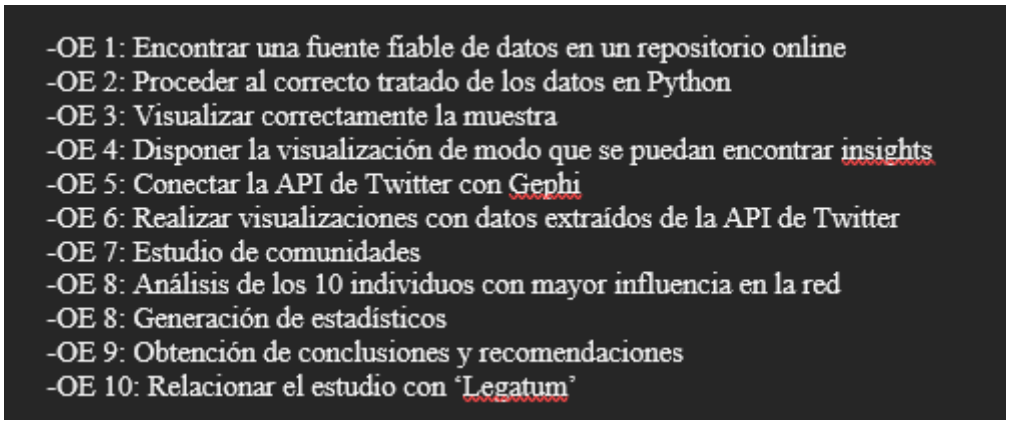
8.2 Conclusiones del trabajo y revisión de objetivos

Las principales conclusiones que se pueden extraer del trabajo se disponen en torno al *dataset* seleccionado y su comportamiento de redes. En este caso, cómo se ha indicado anteriormente, existe una baja conectividad entre la mayoría de los nodos, esto no significa que no sea de relevancia su estudio, pues, son personas en la red.

La solución a esta cuestión oscila entre la incorporación de más información y la disponibilidad de la misma.

Las conclusiones más interesantes, se conforman tras un análisis más exhaustivo de la muestra, en primer lugar, se establece la hipótesis de que el contexto en el que han sido extraídos los datos, guarda algo de relación con el mundo de la música, esto es, fácilmente deducible debido a la presencia de varios artistas de renombre internacional. En cualquier caso, la conclusión más reseñable es que los dos primeros nodos en términos de menciones reciben más inputs negativos que positivos.

En cuanto a los objetivos específicos del trabajo:



Todos fueron completados, exceptuando aquellos relacionados con la API de Twitter (debido al cambio de política de datos de la compañía, ya que, a pesar de tener una cuenta *developer*, no es posible acceder a informaciones fructíferas para este estudio a través de una cuenta de uso gratuito). Por otra parte, no ha sido posible relacionar el estudio de *Legatum*, motivo de mi otro TFG, el cuál ha sido una idea de negocio estrechamente relacionada con la visualización de datos y el manejo de grandes volúmenes de información. Esta idea nace repentinamente, tras sufrir una pérdida familiar muy fuerte a comienzos de año. Se pretendía relacionar los *insights* de este trabajo con '*Legatum*', es por eso, que el set de datos escogido cuenta con una variedad tan alta de etiquetas para sentiment.

8.3 Conclusiones personales

Para concluir este trabajo, me gustaría hacer una pequeña reflexión personal, en primer lugar, los resultados obtenidos en el proyecto me parecen preocupantes, ya que, a pesar de que los sentimiento positivos existan, está demostrado que existe una cultura de toxicisidad en las redes sociales.

Desconozco si esta conducta tóxica viene devengada por la incapacidad de un contacto físico pero ese es uno de los motivos por los que también nace Legatum, para invertir el funcionamiento actual de las redes en el que estamos pendientes de cuestiones ajenas en lugar de las propias.

Como dijo Tirone José González Orama en una de sus canciones antes de fallecer: ‘el ser humano es envidioso y codicioso por placeres objetos’. Quizás este estudio pueda servir como base para prestar mayor atención a los motivos que desencadenan ese tipo de actitudes y que han llevado a personas a perder la vida por acciones transmitidas a través de una pantalla. De hecho, creo que es crucial proteger especialmente a personas con una discapacidad o temprana edad, que no puedan ser conscientes del daño que hacen o reciben hasta que sea demasiado tarde.

Por último, me gustaría señalar que es posible mejorar la calidad de este estudio alterando los parámetros que se han ido mencionando a lo largo del trabajo, pero, por cuestiones personales, no he podido disponer del tiempo para ello.

9. Declaración del uso de IA y Referencias

Declaración de Uso de Herramientas de Inteligencia Artificial Generativa en Trabajos Fin de Grado

ADVERTENCIA: Desde la Universidad consideramos que ChatGPT u otras herramientas similares son herramientas muy útiles en la vida académica, aunque su uso queda siempre bajo la responsabilidad del alumno, puesto que las respuestas que proporciona pueden no ser veraces. En este sentido, NO está permitido su uso en la elaboración del Trabajo fin de Grado para generar código porque estas herramientas no son fiables en esa tarea. Aunque el código funcione, no hay garantías de que metodológicamente sea correcto, y es altamente probable que no lo sea.

Por la presente, yo, Javier Baucells González estudiante de E2 + Analytics de la Universidad Pontificia Comillas al presentar mi Trabajo Fin de Grado titulado Plan de Negocio: Legatum, declaro que he utilizado la herramienta de Inteligencia Artificial Generativa ChatGPT u otras similares de IAG de código sólo en el contexto de las actividades descritas a continuación:

1. **Referencias:** Usado conjuntamente con otras herramientas, como Science, para identificar referencias preliminares que luego he contrastado y validado.
2. **Metodólogo:** Para descubrir métodos aplicables a problemas específicos de investigación.
3. **Sintetizador y divulgador de libros complicados:** Para resumir y comprender literatura compleja.
4. **Traductor:** Para traducir textos de un lenguaje a otro.

Afirmo que toda la información y contenido presentados en este trabajo son producto de mi investigación y esfuerzo individual, excepto donde se ha indicado lo contrario y se han dado los créditos correspondientes (he incluido las referencias adecuadas en el TFG y he explicitado para que se ha usado ChatGPT u otras herramientas similares). Soy consciente de las implicaciones académicas y éticas de presentar un trabajo no original y acepto las consecuencias de cualquier violación a esta declaración.
Fecha: 06/06/2024
Firma: Javier Baucells González

REFERENCIAS BIBLIOGRAFICAS

SAHAYAK, Varsha; SHETE, Vijaya; PATHAN, Apashabi. Sentiment analysis on twitter data. *International Journal of Innovative Research in Advanced Engineering (IJIRAE)*, 2015, vol. 2, no 1, p. 178-183.

MURZONE, Farid (2020). Procesamiento de texto para NLP1: Tokenización. https://medium.com/escueladeinteligenciaartificial/procesamiento-de-texto-para-nlp-1-tokenizaci%C3%B3n-4d533f3f6c9b

GEPHI OFFICIAL TUTORIALS
https://gephi.org/users/quick-start/
https://gephi.org/users/tutorial-visualization/

https://gephi.org/users/tutorial-layouts/

BORGATTI, Stephen P., et al. Network analysis in the social sciences. *science*, 2009, vol. 323, no 5916, p. 892-895.

FREEMAN, Linton, et al. The development of social network analysis. *A Study in the Sociology of Science*, 2004, vol. 1, no 687, p. 159-167.

THANGARAJ, M.; AMUTHA, S. Mgephi: Modified gephi for effective social network analysis. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 2018, vol. 1, no 1, p. 39-50.

CRIADO, J. Ignacio; VILLODRE, Julián. Comunicando datos masivos del sector público local en redes sociales. Análisis de sentimiento en Twitter. *Profesional de la información/Information Professional*, 2018, vol. 27, no 3, p. 624-632.

AMAT, Carlos Benito. Análisis y visualización de redes con Gephi. *Redes. Revista hispana para el análisis de redes sociales*, 2014, vol. 25, no 1, p. 201-209.

CARDOSO, Alejandra Carolina, et al. Minería de opiniones: análisis de sentimientos en una red social. En *XXI Workshop de Investigadores en Ciencias de la Computación (WICC 2019, Universidad Nacional de San Juan).* 2019.

KUZ, Antonieta; FALCO, Mariana; GIANDINI, Roxana. Social network analysis: a practical case study. *Computación y Sistemas*, 2016, vol. 20, no 1.

https://data.world/crowdflower/sentiment-analysis-in-text

https://developer.x.com/en/portal/dashboard

Link otros archivos: https://drive.google.com/drive/folders/1KmBoqChmnxWsFhH93ghY8UNhroet0DeE?usp=drive_link

ANEXO

LINK RESTO DE ARCHIVOS

https://drive.google.com/drive/folders/1KmBoqChmnxWsFhH93ghY8UNhroet0DeE?usp=drive_link

Código de Python completo

```
#Visualizar datos

import pandas as pd
import matplotlib.pyplot as plt

# Cargar el dataset
df = pd.read_csv('text_emotion.csv')

# Mostrar número total de tweets
total_tweets = df.shape[0]
print(f'Número total de tweets: {total_tweets}')

# Mostrar número total de autores
total_authors = df['author'].nunique()
print(f'Número total de autores: {total_authors}')

# Mostrar número total de menciones
# Contar el número de menciones (@) en la columna 'content'
total_mentions = df['content'].str.count('@').sum()
print(f'Número total de menciones: {total_mentions}')

# Crear un gráfico para representar las estadísticas
labels = ['Total Tweets', 'Total Authors', 'Total Mentions']
values = [total_tweets, total_authors, total_mentions]

plt.figure(figsize=(10, 6))
colors = ['#4D4D4D', '#7F7F7F', '#BFBFBF']  # Tonos de gris
escalados
bars = plt.bar(labels, values, color=colors)

# Añadir etiquetas de datos en el centro de cada barra
for bar in bars:
    yval = bar.get_height()
    plt.text(bar.get_x() + bar.get_width() / 2, yval / 2,
int(yval), ha='center', va='center', color='white')

plt.xlabel('Categorías')
plt.ylabel('Cantidad')
```

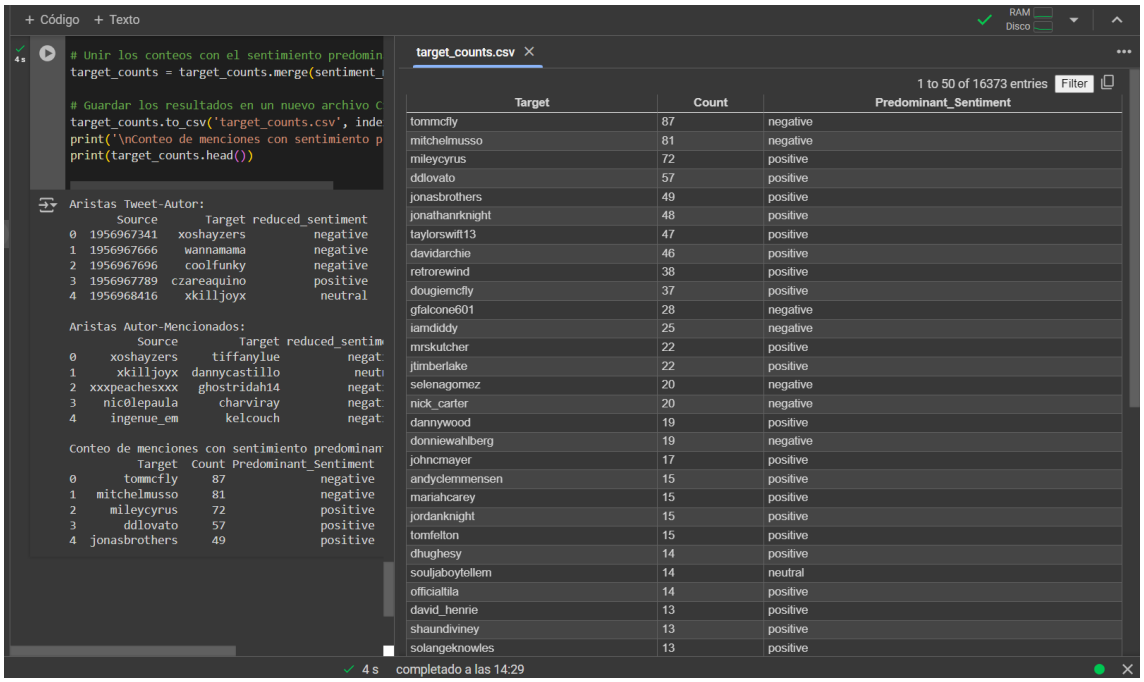
```
plt.title('Estadísticas Básicas del Dataset de Tweets')
plt.show()
import pandas as pd
import matplotlib.pyplot as plt

# Cargar el dataset
ex = pd.read_csv('text_emotion.csv')

# Contar la frecuencia de cada sentimiento
sentiment_counts = ex['sentiment'].value_counts()

# Crear el gráfico de barras
plt.figure(figsize=(12, 8))
colors = ['#4D4D4D', '#7F7F7F', '#BFBFBF', '#E0E0E0', '#F0F0F0']  #
Tonos de gris escalados
bars = sentiment_counts.plot(kind='bar', color=colors, rot=60)

# Añadir etiquetas de datos en el centro de cada barra
for bar in bars.patches:
    yval = bar.get_height()
    plt.text(bar.get_x() + bar.get_width() / 2, yval / 2,
int(yval), ha='center', va='center', color='white')

# Personalizar el gráfico
plt.xlabel('Sentimientos')
plt.ylabel('Cantidad')
plt.title('Distribución de Sentimientos en el Dataset de Tweets')
plt.show()
import pandas as pd
import matplotlib.pyplot as plt

# Cargar el dataset
ex = pd.read_csv('Tweets_reduced.csv')

# Contar la frecuencia de cada sentimiento reducido
reduced_sentiment_counts = ex['reduced_sentiment'].value_counts()

# Crear el gráfico de barras
plt.figure(figsize=(12, 8))
colors = ['#4D4D4D', '#7F7F7F', '#BFBFBF']  # Tonos de gris
bars = reduced_sentiment_counts.plot(kind='bar', color=colors,
rot=60)

# Añadir etiquetas de datos en el centro de cada barra
for bar in bars.patches:
    yval = bar.get_height()
    plt.text(bar.get_x() + bar.get_width() / 2, yval / 2,
int(yval), ha='center', va='center', color='white')
```

```
# Personalizar el gráfico
plt.xlabel('Sentimientos Reducidos')
plt.ylabel('Cantidad')
plt.title('Distribución de Sentimientos Reducidos en el Dataset de
Tweets')
plt.show()

import pandas as pd
df = pd.read_csv('text_emotion.csv')

#Mostrar las primeras filas del dataset
print(df.head())

#Verificar si hay valores nulos
print('Tabla de valores nulos')
print(df.isnull().sum())
print()
#Eliminar filas con valores nulos
df = df.dropna()
import pandas as pd
import re
import matplotlib.pyplot as plt
import seaborn as sns

# Cargar el archivo CSV original con todos los datos
tweets_df = pd.read_csv('text_emotion.csv')  # Reemplaza con la
ruta correcta de tu archivo

# Definir el mapeo de 13 a 5 categorías
mapping = {
    'anger': 'negative',
    'boredom': 'neutral',
    'empty': 'negative',
    'enthusiasm': 'positive',
    'fun': 'positive',
    'happiness': 'positive',
    'hate': 'negative',
    'love': 'positive',
    'neutral': 'neutral',
    'relief': 'positive',
    'sadness': 'negative',
    'surprise': 'surprise',
    'worry': 'negative'
}

# Crear una nueva columna con los valores reducidos
tweets_df['reduced_sentiment'] =
tweets_df['sentiment'].map(mapping)
```

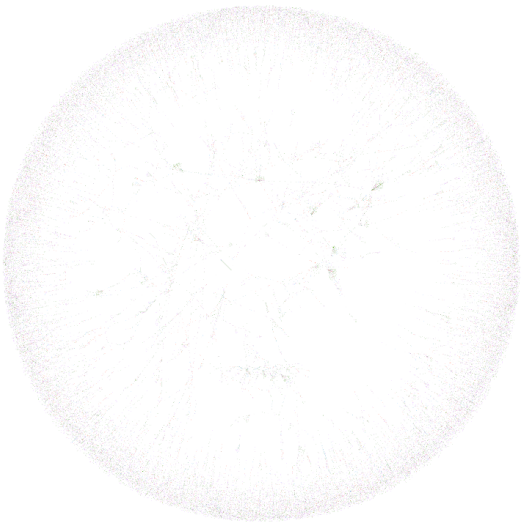
```
# Convertir 'Source' y 'Target' a minúsculas
tweets_df['author'] = tweets_df['author'].str.lower()
tweets_df['usuarios_mencionados'] =
tweets_df['content'].apply(lambda texto: [user.lower() for user in
re.findall(r'@(\w+)', texto)])

# Crear dataframe para las aristas (relación tweet_id -> author)
aristas_tweet_autor = tweets_df[['tweet_id', 'author',
'reduced_sentiment']].drop_duplicates()

# Crear dataframe para las aristas (relación author -> usuarios
mencionados)
aristas_autor_mencionados = tweets_df[['author',
'usuarios_mencionados',
'reduced_sentiment']].explode('usuarios_mencionados').dropna().rese
t_index(drop=True)

# Renombrar columnas para ser compatibles con Gephi
aristas_tweet_autor.rename(columns={'tweet_id': 'Source', 'author':
'Target'}, inplace=True)
aristas_autor_mencionados.rename(columns={'author': 'Source',
'usuarios_mencionados': 'Target'}, inplace=True)

# Guardar el listado de aristas en nuevos archivos CSV
aristas_tweet_autor.to_csv('aristas_tweet_autor.csv', index=False)
aristas_autor_mencionados.to_csv('aristas_autor_mencionados.csv',
index=False)

# Visualizar el listado de aristas final
print('Aristas Tweet-Autor:')
print(aristas_tweet_autor.head())
print('\nAristas Autor-Mencionados:')
print(aristas_autor_mencionados.head())

# Acceder al CSV 'aristas_autor_mencionados.csv' y contar las
ocurrencias de cada valor en la columna 'Target'
edges_data = pd.read_csv('aristas_autor_mencionados.csv')

# Calcular el sentimiento predominante para cada Target
sentiment_mode =
edges_data.groupby('Target')['reduced_sentiment'].agg(lambda x:
x.mode()[0]).reset_index()
sentiment_mode.columns = ['Target', 'Predominant_Sentiment']

# Contar las menciones de cada Target
target_counts = edges_data['Target'].value_counts().reset_index()
target_counts.columns = ['Target', 'Count']

# Unir los conteos con el sentimiento predominante
```

```
target_counts = target_counts.merge(sentiment_mode, on='Target',
how='left')

# Guardar los resultados en un nuevo archivo CSV
target_counts.to_csv('target_counts.csv', index=False)
print('\nConteo de menciones con sentimiento predominante:')
print(target_counts.head())
print()

# Seleccionar los 10 usuarios con más menciones
top_10_users = target_counts.head(10)

# Crear el gráfico
plt.figure(figsize=(12, 8))
sns.barplot(data=top_10_users, x='Count', y='Target',
hue='Predominant_Sentiment', palette='gray')

# Configurar etiquetas y título
plt.xlabel('Número de Menciones')
plt.ylabel('Usuarios Mencionados')
plt.title('Top 10 Usuarios Mencionados y Sentimiento Predominante')
plt.legend(title='Sentimiento Predominante')

# Mostrar el gráfico
plt.show()
```

Vista preliminar de Target_counts.csv

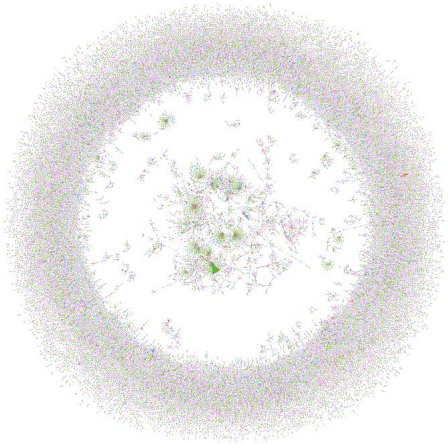


Workspace 1



Workspace 2

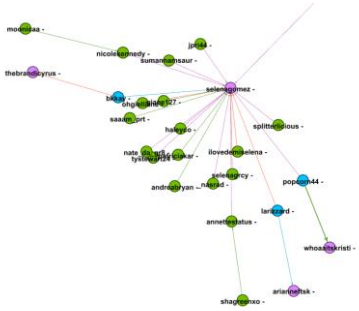
Workspace 3



Distribución de Targets más influyentes

Color azul: David archie, color rojo: mayoría de los targets, color verde: jordan knight.

Otros influencers



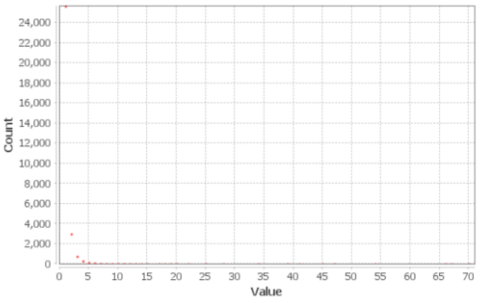
Degree Report

Degree Report

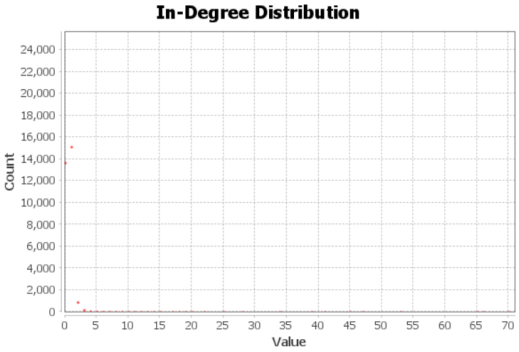
Results:

Average Degree: 0.634

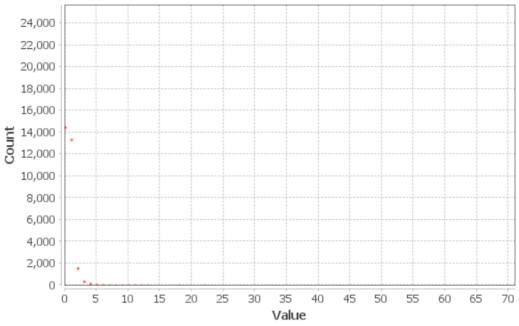
Degree Distribution



In-Degree Distribution



Out-Degree Distribution

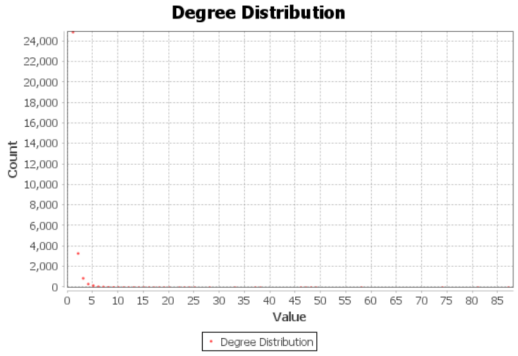


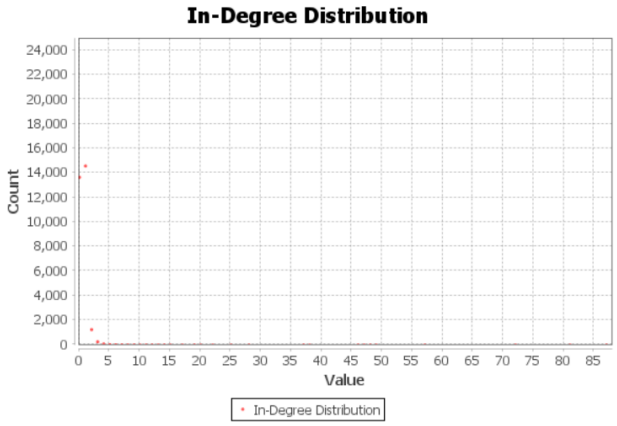
Weighted Degree Report

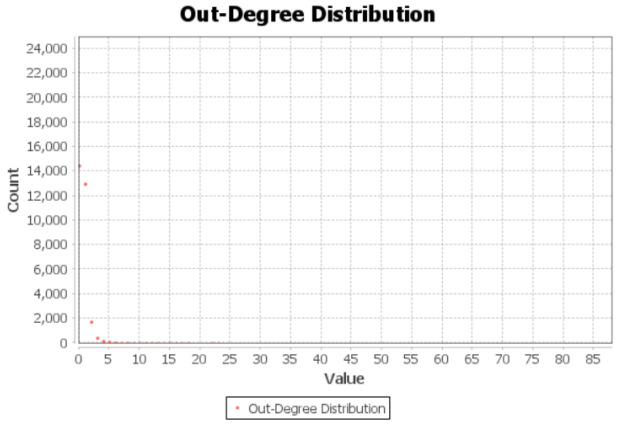
Weighted Degree Report

Results:

Average Weighted Degree: 0.666







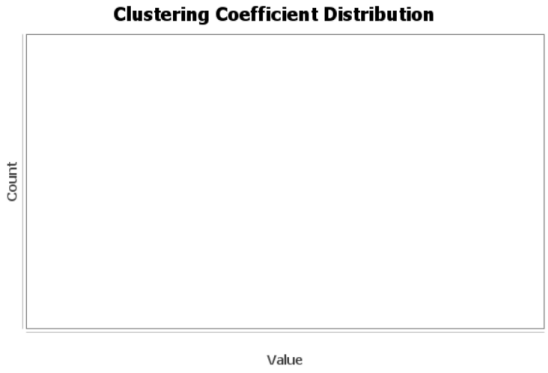
Clustering coefficient

Parameters:

Network Interpretation: directed

Results:

Average Clustering Coefficient: 0.003
The Average Clustering Coefficient is the mean value of individual coefficients.



Connected Components Report

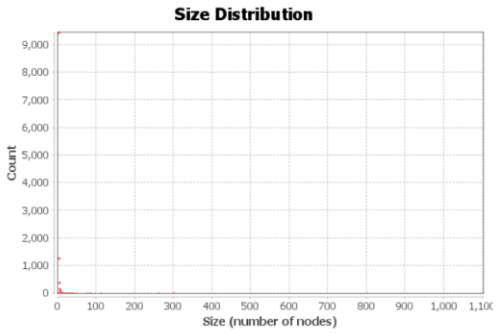
Connected Components Report

Parameters:

Network Interpretation: directed

Results:

Number of Weakly Connected Components: 11617
Number of Strongly Connected Components: 29668



Algorithm:

Robert Tarjan, *Depth-First Search and Linear Graph Algorithms*, in SIAM Journal on Computing 1 (2): 146–160 (1972)

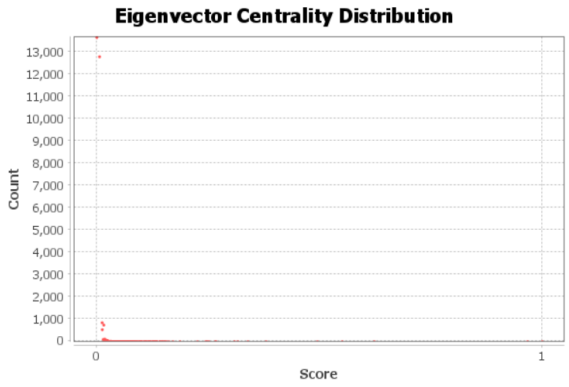
Eigenvector Centrality

Eigenvector Centrality Report

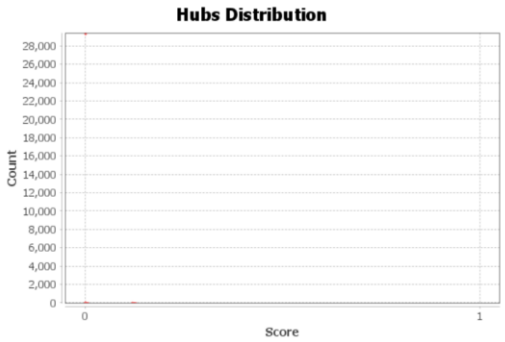
Parameters:

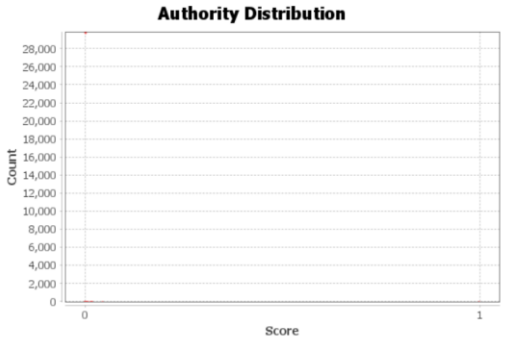
Network Interpretation: directed
Number of iterations: 100
Sum change: 0.743562355531949

Results:



HITS Report





Modularity report

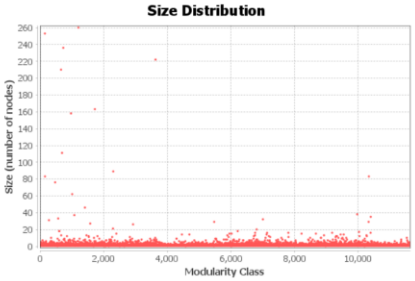
Modularity Report

Parameters:

Randomize: On
Use edge weights: On
Resolution: 1.0

Results:

Modularity: 0.998
Modularity with resolution: 0.998
Number of Communities: 11623



Algorithm:

Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, Etienne Lefebvre, *Fast unfolding of communities in large networks*, in Journal of Statistical Mechanics: Theory and Experiment 2008 (10), P1000

Graph Distance Report

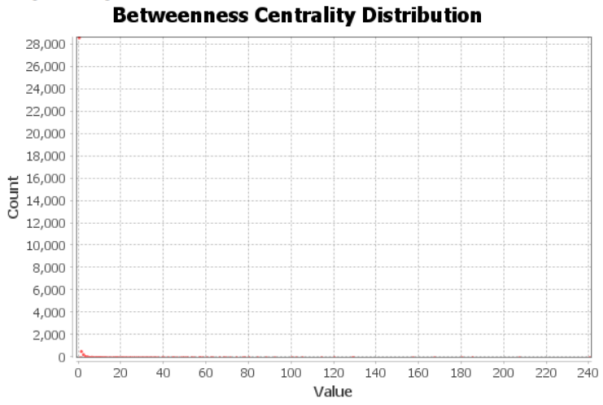
Graph Distance Report

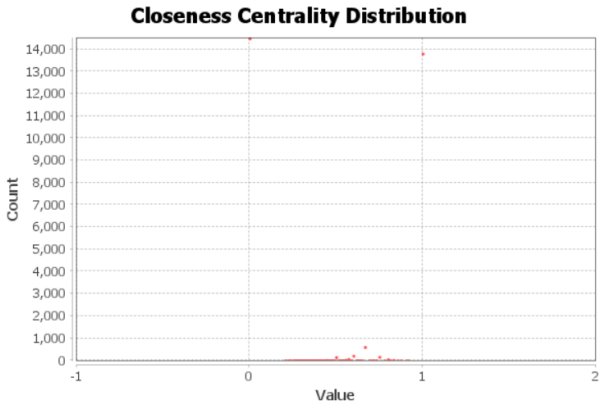
Parameters:

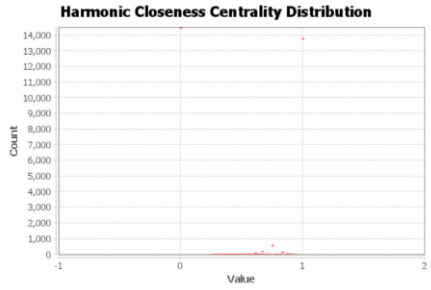
Network Interpretation: directed

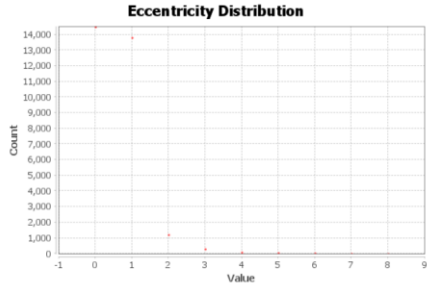
Results:

Diameter: 8
Radius: 0
Average Path length: 1.413093904975204









Algorithm:

Ulrik Brandes, *A Faster Algorithm for Betweenness Centrality*, in Journal of Mathematical Sociology 25(2):163-177, (2001)

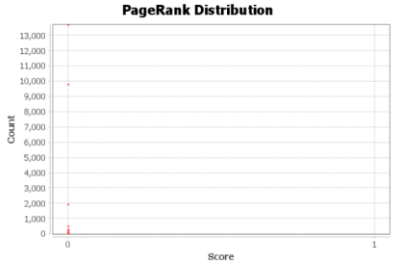
PageRank Report

PageRank Report

Parameters:

Epsilon = 0.001
Probability = 0.85

Results:



Algorithm:

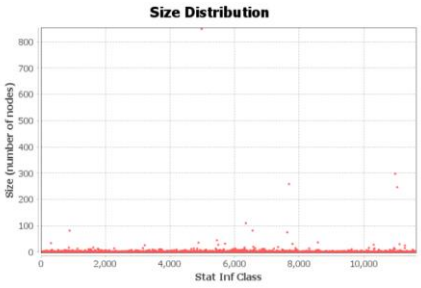
Page, Lawrence and Brin, Sergey and Motwani, Rajeev and Winograd, Terry (1999) *The PageRank Citation Ranking: Bringing Order to the Web.* Technical Report. Stanford InfoLab.

Statistical Inference Report

Statistical Inference Report

Results:

Description Length: 331948.871
Number of Communities: 11618



Algorithm:

Statistical inference of assortative community structures
Lizhi Zhang, Tiago P. Peixoto
Phys. Rev. Research 2 043271 (2020)
https://dx.doi.org/10.1103/PhysRevResearch.2.043271

Bayesian stochastic blockmodeling
Tiago P. Peixoto
Chapter in "Advances in Network Clustering and Blockmodeling," edited by
P. Doreian, V. Batagelj, A. Ferligoj (Wiley, 2019)
https://dx.doi.org/10.1002/9781119483298.ch11