

Representació d'una conversa de Twitter en forma de graf

Autor: Marc Sánchez Fauste

Director: Jordi Planes Cid

Grau en Enginyeria Informàtica

Escola Politècnica Superior

Universitat de Lleida

Treball Final de Grau

Juliol de 2016

Índex

1	Introducció i motivació	7
2	Anàlisi	9
2.1	Requeriments	9
2.2	Casos d'ús	10
2.3	Tecnologia	11
2.3.1	Python	12
2.3.2	JavaScript	13
2.3.3	Django	14
2.3.4	Bootstrap	14
2.3.5	D3	15
3	Disseny	17
3.1	Disseny estàtic	17
3.2	Disseny dinàmic	18
3.2.1	Importar una conversa	18
3.2.2	Mostrar conversa en forma de graf	19
3.3	Reconstrucció d'una conversa	19
4	Test	21
5	Conclusions	27
	Bibliografia	29

Índex de figures

2.1	Diagrama de casos d'ús del sistema	12
3.1	Diagrama de classes de l'aplicació	17
3.2	Diagrama de seqüència del cas d'ús importar una conversa	18
3.3	Diagrama de seqüència del cas d'ús mostrar conversa en forma de graf	19
4.1	Pantalla on es mostren les converses importades al sistema	23
4.2	Pantalla on es mostra una conversa	24
4.3	Conversa de la figura 4.2 en forma de graf - llegenda sentiment	25
4.4	Conversa de la figura 4.2 en forma de graf - llegenda favorits	26

Capítol 1

Introducció i motivació

El propòsit d'aquest projecte és crear un sistema capaç de representar qualsevol conversa de la plataforma Twitter en forma de graf, tenint la opció d'escollir entre diferents criteris a utilitzar a l'hora de generar-lo, i una vegada generat, poder veure i analitzar les relacions que s'estableixen entre els tweets que formen la conversa. Un possible exemple seria veure si els usuaris que estan responenent al tweet inicial estan recolzant-lo o atacant-lo.

L'objectiu final és crear una plataforma web que sigui capaç d'importar converses de Twitter a partir de l'identificador d'un tweet arrel, i un cop importades, poder modificar des d'un panell el sentiment que té associat cada tweet (en contra, neutral o a favor). En un primer moment la detecció d'aquest sentiment es realitzarà de forma automàtica, utilitzant una eina creada per a detectar el sentiment d'un text, que ha estat desenvolupada pel Grup de Recerca en Intel·ligència Artificial de la Escola Politècnica Superior de la Universitat de Lleida.

Tot i que existeixen moltes xarxes socials en l'actualitat, Twitter és una de les més utilitzades quan es tracta de compartir i criticar notícies rellevants. Els ciutadans responen a les notícies i esdeveniments a Twitter, i això es pren amb freqüència com un indicador de l'interès social. Per exemple, això es pot observar en els esforços dels investigadors en ciències socials i la intel·ligència artificial per utilitzar eines automàtiques per analitzar les tendències en les respostes dels usuaris de Twitter a certs esdeveniments, i fins i tot, l'ús de dades de Twitter per provar les opinions més predominants en la política, o fins i tot, per predir resultats pel que fa a les eleccions governamentals. L'elecció de Twitter com un objectiu i una eina d'investigació en considerar diferents xarxes socials no és una decisió arbitrària, sinó que es basa en dos fets principals: la mida i l'obertura. Pel que fa a la mida, Twitter té una audiència activa de més de 320 milions d'usuaris al mes (a desembre de 2015), amb un màxim d'1,3 mil milions d'usuaris registrats, i una freqüència de piulades d'aproximadament 6.000 tweets per segon. En quant a l'obertura, Twitter és una xarxa social oberta, fàcil d'utilitzar en investigació gràcies a dues decisions de disseny: en primer lloc, tots els tweets, llevat que s'indiqui el contrari, estan oberts per a ser llegits per qualsevol persona, i en segon lloc, des dels seus primers dies, Twitter té una extensa API pública que permet als desenvolupadors interactuar amb els servidors de Twitter, i facilitar l'extracció de grans quantitats de dades de forma senzilla.

Un altre fet que revela que Twitter és àmpliament conegut i utilitzat per la societat, és que molts

mitjans de comunicació conviden als seus espectadors a donar la seva opinió sobre el tema que s'està debatent en directe. L'únic que han de fer és escriure un tweet manifestant la seva opinió i afegir una etiqueta concreta. D'altres vegades són els mateixos usuaris de Twitter els que inicien una discussió a la plataforma i inviten als altres usuaris a donar la seva opinió sobre el tema. És per això, que si cerquem tweets sobre algun tema polèmic, ja sigui filtrant els resultats a partir d'una etiqueta o d'una sèrie de paraules representatives, podem trobar petits debats que han anat sorgint, ja que com que tothom pot opinar lliurement, la diversitat d'opinions és molt àmplia i és fàcil trobar converses que s'han generat a partir de la opinió d'un usuari. En aquestes converses que es van formant ens podem trobar respostes que recolzen la idea de la persona que ha iniciat la discussió, respostes que defenen un altre punt de vista o que directament hi estan totalment en contra.

Una conversa es pot representar de manera natural en forma de graf dirigit, considerant que cada node representa un tweet de la conversa i que les arestes representen les relacions que s'estableixen entre els diferents tweets. En aquest cas una aresta indicarà que un tweet és resposta d'un altre tweet. Utilitzant aquest format es podrà veure a simple vista quins són els tweets de la conversa que més respostes han obtingut, i per tant, poden ser considerats més rellevants. És per aquest motiu que s'ha escollit aquest format per a representar les converses.

Capítol 2

Anàlisi

2.1 Requeriments

Volem crear un sistema que sigui capaç d'importar converses de la xarxa social Twitter i emmagatzemar-les en una base de dades local, per a poder tractar-les posteriorment sense haver de dependre de dades externes.

La forma en que seran importades les converses consistirà en obtenir recursivament totes les respostes d'un tweet arrel fins haver explorat tota la discussió.

Tots els tweets tindran associada una variable entera que representarà numèricament el sentiment que té associat aquell tweet en particular. El criteri serà el següent:

- Un nombre negatiu voldrà dir que aquell tweet té associat un sentiment negatiu, ja sigui perquè està en contra del tweet que està responent, o simplement perquè està transmetent una opinió negativa o està en desacord sobre el tema que està parlant.
- El 0 indicarà que el sentiment és neutre, o dit d'altra manera, que no es posiciona ni a favor ni en contra.
- Un nombre igual o superior a 1 indica un sentiment positiu o que està a favor amb el tweet que està responent.

Evidentment, aquesta variable també representarà el nivell del sentiment, per tant, un tweet amb un sentiment de -50 serà considerat més negatiu que, per exemple, un tweet que tingui associat un sentiment de -10 . El mateix passa amb els sentiments positius, com més s'allunyin del 0 , més positius seran considerats.

Quan importem un tweet, aquest passarà per un analitzador que en determinarà el sentiment de forma automàtica.

El sistema ha de permetre editar el sentiment d'un tweet a partir d'una interfície web que mostri tots els tweets associats a una conversa.

El sistema ha de ser capaç de representar en forma de graf dirigit les converses que s'han importat prèviament. Anem a veure quines característiques ha de tenir el graf generat:

- Els nodes del graf representaran tots els tweets de la conversa. Així cada node representarà un tweet de la conversa.
- Les arestes del graf indicaran les relacions que tenen els nodes. En aquest cas que un node és una resposta d'un altre node.
 - Un node pot tenir de 0 a $N-1$ respostes.
 - Un node pot respondre a 0 o 1 nodes.
- Els nodes del graf seran colorejats segons la llegenda que s'escolleixi:
 - El color de cada node depèn del sentiment que té associat.
 - El color depèn del nombre de retweets que té el node.
 - El color es determina a partir del nombre de favorits que té cada node.
- Els nodes aniran numerats de 0 a $N-1$, on el número indicarà la posició que ocupen a la conversa. Així el 0 serà el tweet arrel, l' 1 serà la primera resposta i així successivament.

El sistema permetrà que hi puguin haver diferents comptes d'usuari.

Un usuari registrat podrà veure i modificar les converses que hagi importat anteriorment.

Un usuari registrat podrà veure les converses que hagin importat els altres usuaris.

Un usuari no identificat podrà veure totes les converses que té emmagatzemades el sistema, però no les podrà modificar.

S'haurà de crear una interfície capaç de comunicar-se amb l'API de Twitter i demanar-li les dades que siguin necessàries per tal que el sistema pugui dur a terme la funcionalitat aquí esmentada.

Com que hi ha dades que el sistema no serà capaç d'obtenir a partir de l'API de Twitter, s'haurà de buscar una alternativa per tal d'obtenir aquestes dades.

2.2 Casos d'ús

Com es pot veure a la figura 2.1, tenim 5 possibles casos d'ús. Anem a analitzar-los:

- **Importar una conversa**
 - **Actors:** Usuari registrat, Twitter.
 - **Propòsit:** Importar una conversa al sistema a partir d'un tweet arrel.

- **Descripció:** Un usuari registrat es connecta al sistema i procedeix a importar una conversa indicant l'id que té a la xarxa social Twitter el tweet que és vol considerar com a arrel de la conversa. El sistema realitza totes les consultes pertinents a Twitter per tal d'aconseguir la conversa completa. Una vegada s'ha acabat d'executar el procediment, la conversa estarà disponible a la base de dades del sistema i l'usuari serà notificat.

- **Cercar una conversa importada**

- **Actors:** Usuari invitat, usuari registrat.
- **Propòsit:** Cercar una conversa a la base de dades del sistema.
- **Descripció:** Un usuari consulta la llista de converses que han estat importades al sistema. El sistema mostra una llista amb totes les converses que han estat importades. De cada conversa es mostra el contingut del tweet arrel.

- **Mostrar conversa en forma de graf**

- **Actors:** Usuari invitat, usuari registrat.
- **Propòsit:** Visualitzar una conversa importada al sistema en forma de graf.
- **Descripció:** Un usuari es connecta al sistema i cerca una conversa. Una vegada ha trobat la conversa que vol visualitzar en forma de graf, selecciona la opció generar graf i el sistema genera un graf que representa aquesta conversa.

- **Mostrar conversa**

- **Actors:** Usuari invitat, usuari registrat.
- **Propòsit:** Visualitzar una conversa importada al sistema.
- **Descripció:** Un usuari es connecta al sistema i cerca una conversa. Una vegada ha trobat la conversa que vol visualitzar, selecciona la opció mostrar i el sistema li mostra la conversa completa.

- **Editar sentiment d'un tweet**

- **Actors:** Usuari registrat.
- **Propòsit:** Modificar el sentiment associat a un tweet.
- **Descripció:** Un usuari es connecta al sistema, cerca una conversa i la mostra. Una vegada es troba dins de la conversa, cerca el tweet al qual vol editar el sentiment i procedeix a seleccionar quin sentiment ha de tenir el tweet. Una vegada s'ha realitzat aquesta acció la conversa és actualitzada i es pot repetir el cas d'ús per a qualsevol altre tweet de la conversa que s'està mostrant.

2.3 Tecnologia

En aquesta secció veurem la tecnologia que s'ha utilitzat a l'hora de realitzar el projecte.

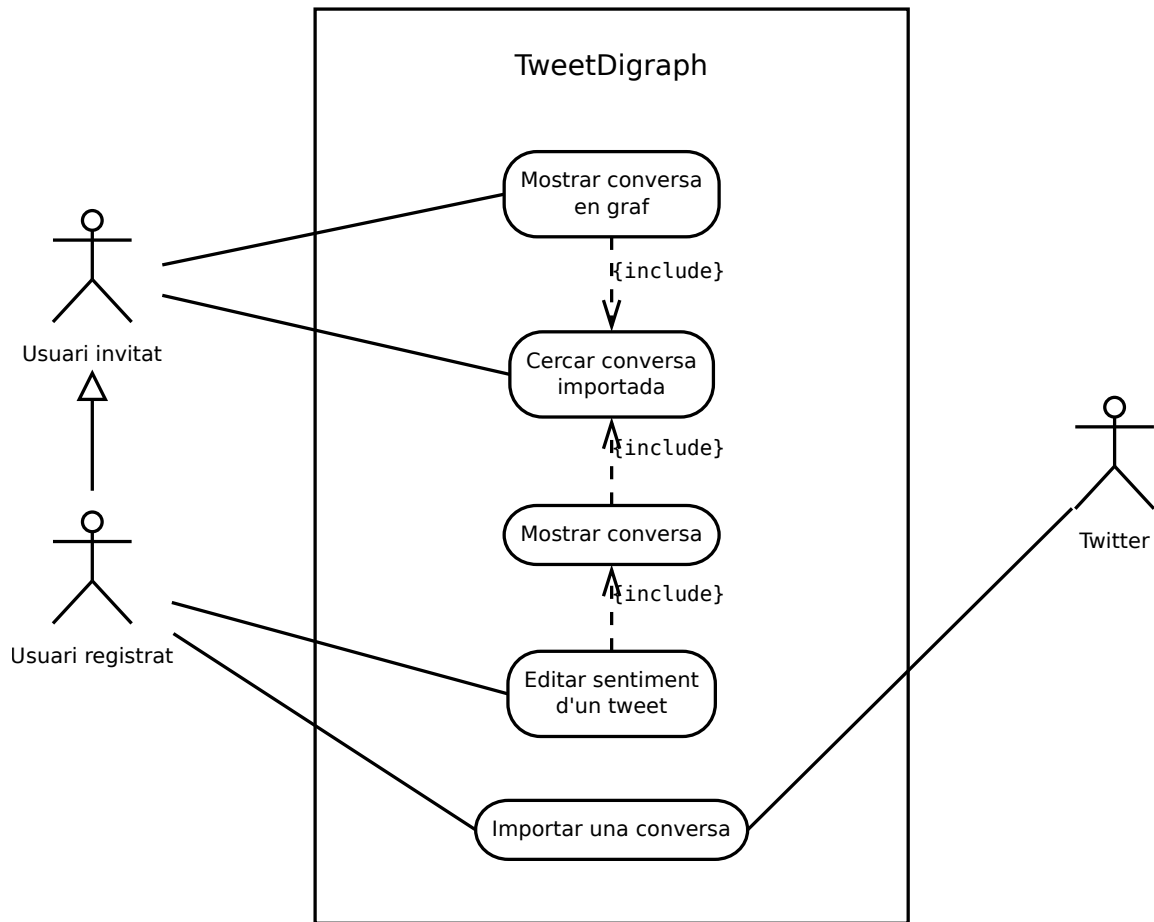


Figura 2.1: Diagrama de casos d'ús del sistema

2.3.1 Python

Python és un llenguatge de programació d'alt nivell i propòsit general molt utilitzat. Va ser creat per Guido van Rossum l'any 1991. La seva filosofia de disseny busca llegibilitat en el codi i la seva sintaxi permet als programadors expressar conceptes en menys línies de codi del que seria possible en llenguatges com C. També proveeix estructures per permetre programes més entenedors tant a petita com a gran escala.

Python suporta diversos paradigmes de programació, incloent-hi programació orientada a objectes, imperativa i també funcional o procedimental. Presenta un sistema dinàmic i una gestió de la memòria automàtica, i té una gran i exhaustiva biblioteca estàndard.

Com altres llenguatges de programació dinàmics, Python és usat sovint com a un llenguatge script, però també es fa servir en una àmplia gamma de contextos no-script. Utilitzant eines desenvol-

padres per tercers com Py2exe, cx_Freeze o Pyinstaller, el codi Python pot ser reduït a programes executables independents. Existeixen intèrprets de Python per molts sistemes operatius diferents.

La implementació principal del Python, coneguda com a CPython, està escrita en C, satisfent l'estàndard C89. El CPython compila els programes Python en un bytecode intermedi, que posteriorment s'executa per la màquina virtual. Es distribueix amb una llibreria estàndard força gran que està escrita utilitzant una barreja de C i Python. Té versions per moltes plataformes, incloent Microsoft Windows i la major part dels sistemes basats en Unix. Des de la seva concepció va ésser dissenyat per ser multiplataforma; el seu ús i desenvolupament en plataformes esotèriques com Aomeba, juntament amb altres més convencionals com Unix i Mac OS, ha ajudat en aquest aspecte. El 2009 Google va començar un projecte anomenat Unladen Swallow amb l'objectiu d'incrementar la velocitat de l'interpretador Python unes cinc vegades.

2.3.2 JavaScript

JavaScript (abreujat com JS) és un llenguatge de programació interpretat, dialecte de l'estàndard ECMAScript. Es defineix com orientat a objectes, basat en prototips, imperatiu, dèbilment tipat i dinàmic.

S'utilitza principalment al costat del client (client-side), implementat com a part d'un navegador web permetent millores en la interfície d'usuari i pàgines web dinàmiques encara que existeix el JavaScript al costat del servidor (Server-side JavaScript o SSJS). El seu ús en aplicacions externes a la web, per exemple en documents PDF, aplicacions d'escriptori (majoritàriament widgets) és també significatiu.

JavaScript es va dissenyar amb una sintaxi similar a C, encara que adopta noms i convencions del llenguatge de programació Java. No obstant això, Java i JavaScript tenen semàntiques i propòsits diferents.

Tots els navegadors moderns interpreten el codi JavaScript integrat en les pàgines web. Per interactuar amb una pàgina web es proveeix al llenguatge JavaScript una implementació del Document Object Model (DOM).

Tradicionalment s'utilitzava en pàgines web HTML per realitzar operacions i únicament en el marc de l'aplicació client, sense accés a funcions del servidor. Actualment és àmpliament utilitzat per enviar i rebre informació del servidor juntament amb ajuda d'altres tecnologies com AJAX. JavaScript s'interpreta en l'agent d'usuari al mateix temps que les sentències es van descarregant junt amb el codi HTML.

Des del llançament al juny de 1997 de l'estàndard ECMAScript 1, han existit les versions 2, 3 i 5, que és la més usada actualment (la 4 es va abandonar). Al juny de 2015 es va tancar i publicar la versió ECMAScript 6.

2.3.3 Django

El sistema s'ha implementat en forma d'aplicació web, utilitzant Django. Aquest és un framework de desenvolupament web de codi obert, escrit en Python i que es basa en el patró de disseny conegut com Model-vista-controlador.

A l'igual que Ruby on Rails, un altre popular framework de codi obert, Django es va estar utilitzant en producció durant un temps abans de ser lliberat al públic. Va ser desenvolupat per Adrian Holovaty, Simon Willison, Jacob Kaplan-molsa i Wilson Miner mentre treballaven a World Online, i originalment es va utilitzar per a administrar 3 llocs web de notícies: The Lawrence Journal-World, lawrence.com i KUsports.com.

Els orígens de Django en l'administració de pàgines de notícies són evidents en el seu disseny, ja que proporciona una sèrie de característiques que faciliten el desenvolupament ràpid de pàgines orientades a continguts. Per exemple, en lloc de requerir que els desenvolupadors escriguin controladors i vistes per a les àrees d'administració de la pàgina, Django proporciona una aplicació incorporada per administrar els continguts, que es pot incloure com a part de qualsevol pàgina feta amb Django i que pot administrar diverses pàgines fetes amb Django a partir d'una mateixa instal·lació. L'aplicació administrativa permet la creació, actualització i eliminació d'objectes de contingut, portant un registre de totes les accions realitzades sobre cadascun, i proporciona una interfície per administrar els usuaris i els grups d'usuaris (incloent una assignació detallada de permisos).

He decidit utilitzar aquest framework, ja que és una eina que ja havia utilitzat anteriorment i et permet crear una aplicació web de forma fàcil i ordenada. El patró en el que es basa permet tenir el codi de l'aplicació ben organitzat. Un dels altres avantatges d'utilitzar el framework, tal com s'ha comentat anteriorment, és que incorpora un panell administratiu des d'on es poden gestionar les instàncies dels models que utilitza l'aplicació. Aquesta funcionalitat facilita el desenvolupament i la gestió de l'aplicació.

2.3.4 Bootstrap

Bootstrap és un framework o conjunt d'eines de codi obert que són útils a l'hora de realitzar el disseny gràfic d'una pàgina web. Conté plantilles de disseny amb tipografia, formularis, botons, quadres, menús de navegació i altres elements de disseny basats en HTML i CSS, així com extensions de JavaScript.

Bootstrap té un suport relativament incomplet per HTML5 i CSS 3, però és compatible amb la majoria dels navegadors web. La informació bàsica de compatibilitat de llocs web o aplicacions està disponible per a tots els dispositius i navegadors. Existeix un concepte de compatibilitat parcial que fa disponible la informació bàsica d'un lloc web per a tots els dispositius i navegadors. Per exemple, les propietats introduïdes en CSS3 per les cantonades arrodonides, gradients i ombres són utilitzades per Bootstrap tot i la manca de suport en navegadors antics. Això estén la funcionalitat de l'eina, però no és requerida per al seu ús.

Des de la versió 2.0 també suporta dissenys sensibles. Això vol dir que el disseny gràfic de la pàgina s'ajusta dinàmicament, tenint en compte les característiques del dispositiu usat (Ordinadors,

tauletes, telèfons mòbils).

Bootstrap és de codi obert i està disponible a GitHub. Els desenvolupadors estan motivats a participar en el projecte i a fer les seves pròpies contribucions a la plataforma.

He decidit utilitzar aquesta eina, ja que està molt ben documentada, i per tant, és fàcil d'incorporar al nostre codi, i perquè li dona un aspecte agradable a la interfície gràfica de la nostra web, sense que haguem de dissenyar nosaltres l'estil de tots els elements que apareixen a les diferents seccions de la web.

2.3.5 D3

D3 (o D3.js) és una llibreria escrita en JavaScript que permet visualitzar dades utilitzant els estàndards web.

Aquesta llibreria inclou eines per representar tot tipus de dades en diferents formats, però el que ens interessa pel nostre projecte és representar dades en forma de graf dirigit, per tant, només s'ha utilitzat la part referent a representació de dades en forma de graf.

El que permet l'eina és que a partir d'un graf base que nosaltres definim, l'usuari sigui capaç de modificar aquest graf afegint o eliminant nodes i arestes. El que fa el nostre sistema és generar i mostrar el graf que representa la conversa i dóna la possibilitat a l'usuari de modificar el graf generat sense que es desin els canvis al sistema.

Capítol 3

Disseny

3.1 Disseny estàtic

Com és pot veure a la figura 3.1, la classe principal de l'aplicació és Tweet. Aquesta és l'encarregada d'emmagatzemar totes les converses que s'importen al sistema.

Tots els tweets de la conversa són emmagatzemats d'un en un al sistema, i a part, també es guarda quin és el tweet arrel de cada conversa utilitzant la classe UserConversation. D'aquesta manera és possible recuperar la conversa posteriorment a partir d'agar el tweet arrel i obtenir els tweets que són resposta d'aquest de forma recursiva. A la secció 3.3 s'explica aquest procediment de forma més detallada.

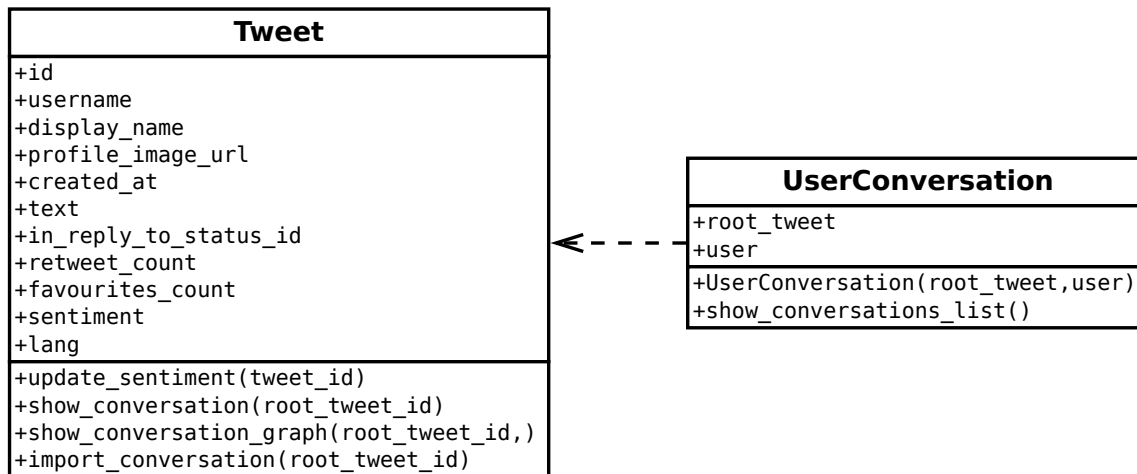


Figura 3.1: Diagrama de classes de l'aplicació

3.2 Disseny dinàmic

En aquesta secció veurem el diagrama de seqüència dels casos d'ús més importants de l'aplicació.

3.2.1 Importar una conversa

Com es pot observar a la figura 3.2 l'usuari s'ha d'identificar al sistema i una vegada s'ha identificat correctament, procedir a importar la conversa desitjada indicant l'id que té a Twitter el tweet que es vol agafar com arrel de la conversa. Una vegada s'ha especificat al sistema la conversa que ha d'importar, aquest procedeix a realitzar la consulta necessària a Twitter per a obtenir els diferents tweets que formen la conversa i una vegada Twitter li retorna aquesta informació, procedeix a emmagatzemar-la a la base de dades i informar a l'usuari que la conversa ha estat desada correctament.

Un fet a destacar és que l'API de Twitter no proporciona un mètode per obtenir les respostes que té un tweet. És per aquest motiu que he hagut d'utilitzar tècniques de web scraping¹ per a obtenir aquesta informació. Una vegada s'ha obtingut aquesta informació ja es pot procedir a demanar els tweets pertinents a l'API de Twitter.

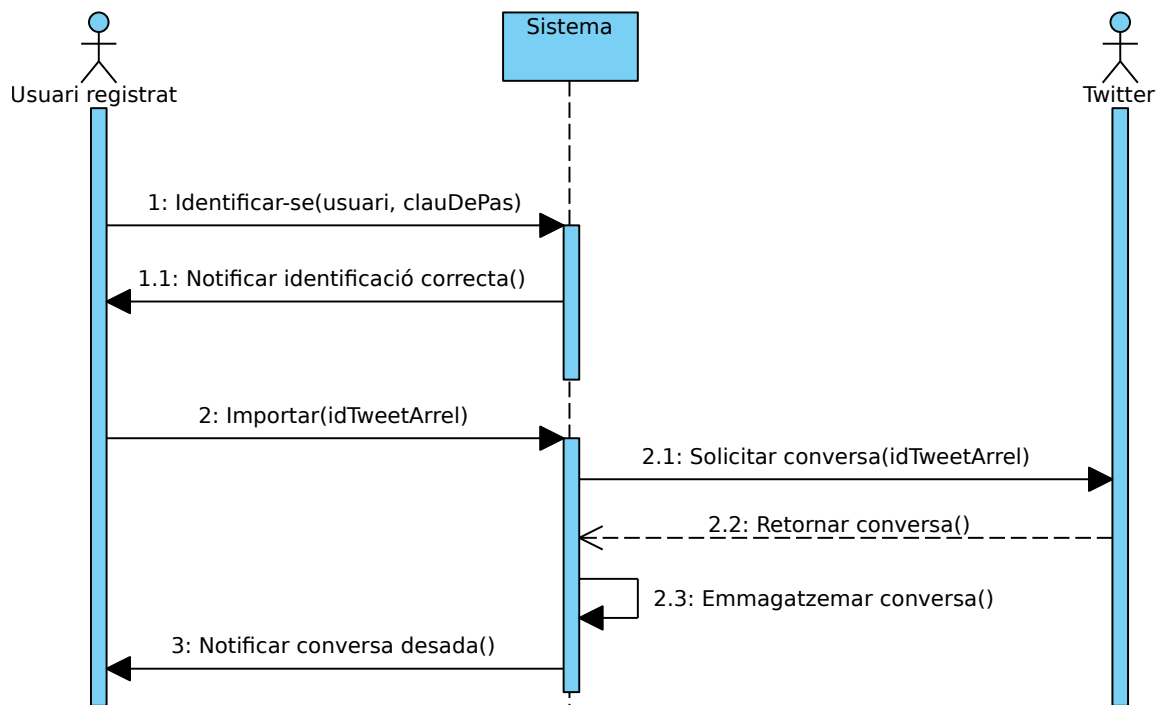


Figura 3.2: Diagrama de seqüència del cas d'ús importar una conversa

¹És una tècnica utilitzada mitjançant programes de software per a extreure informació de pàgines web. Aquests programes simulen la navegació web d'un humà ja sigui utilitzant el protocol HTTP o incrustant un navegador web a una aplicació.

3.2.2 Mostrar conversa en forma de graf

Com es pot veure a la figura 3.3, aquest és el procediment que realitza el sistema quan un usuari vol generar el graf que representa una conversa concreta. El primer que ha de fer l'usuari és veure el llistat de converses que han estat importades al sistema, i determinar a partir de quina conversa vol generar el graf. Una vegada ha determinat quina conversa vol utilitzar, procedeix a sol·licitar el graf al sistema. El sistema recupera la conversa (veure secció 3.3), genera el graf i finalment el mostra a l'usuari.

Si es volgués realitzar el cas d'ús mostrar conversa, únicament s'hauria de suprimir l'acció generar graf. És a dir, una vegada el sistema ha recuperat la conversa la mostraria a l'usuari, saltant-se el procediment que genera el graf.

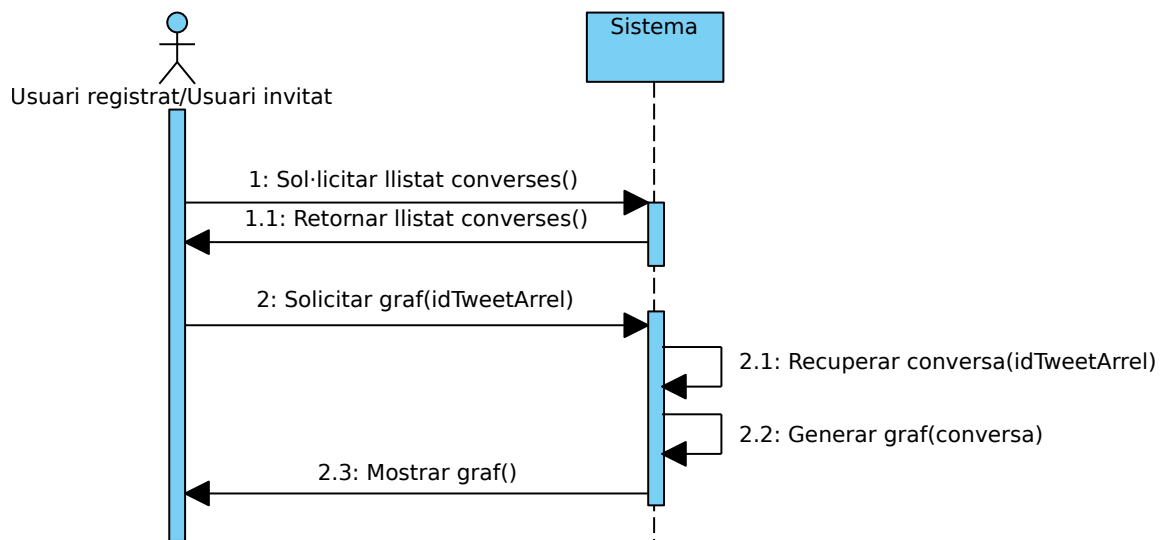


Figura 3.3: Diagrama de seqüència del cas d'ús mostrar conversa en forma de graf

3.3 Reconstrucció d'una conversa

Un aspecte important a destacar de la implementació del sistema, és la manera en que una conversa és recuperada de la base de dades local. Aquest és un procediment que es duu a terme cada vegada que se li mostra una conversa a un usuari, ja sigui en forma de graf o mostrant el text de totes les respostes.

Anem a veure i comentar l'algorisme encarregat de realitzar aquesta funció:

```

1 def get_conversation(root_tweet):
2     stack = [root_tweet]
3     conversation = []
4     while (len(stack)):
  
```

```

5     tweet = stack.pop()
6     conversation.append(tweet)
7     replies = Tweets.filter(
8         in_reply_to_status_id=tweet.id)
9     for reply in replies:
10        stack.append(reply)
11    return conversation

```

Com es pot veure el que fa aquest algorisme és obtenir les respostes de forma recursiva i emmagatzemar-les en una llista. Tot i que de forma natural aquest algorisme té forma recursiva, en aquest cas ha estat traduït a forma iterativa per a què la seva execució sigui més eficient, i també per evitar desbordaments de pila en el cas que la conversa tingui moltes respostes.

Una vegada el sistema ha recuperat la conversa de la base de dades, aquesta ja està a punt per a ser mostrada a l'usuari. En aquest punt podem distingir dos casos diferents, depenent de la opció que hagi escollit prèviament l'usuari:

- **Mostrar la conversa:** el que farà el sistema serà recórrer tots els tweets de la conversa i anar-los mostrant a l'usuari.
- **Representar la conversa en forma de graf:** el que farà el sistema serà generar un graf que contingui N nodes numerats de 0 a $N-1$, on N és el número de tweets que conté la conversa i on el node 0 és el tweet arrel i la resta són respostes. L'ordre en que es numeren els tweets ve determinat en funció de la posició que ocupen dins la conversa.

Les arestes del graf representaran les relacions que hi ha entre els diferents nodes. En aquest cas, una aresta entre dos nodes significarà que un node està responent a un altre, per tant, una aresta que vagi des del node 1 al 0 indicarà que el node 1 és una resposta al node 0 . Cal destacar que un node només pot ser resposta d'un únic node, però un node pot contenir de 0 a $N-1$ respostes.

Una vegada s'han determinat els nodes i les arestes que definiran el graf que s'ha de mostrar a l'usuari, també és procedirà a aplicar la llegenda per defecte, que determina que el color dels nodes va en funció del sentiment que tenen associat. L'usuari té la possibilitat d'escollir entre diverses llegendes tal i com s'indica a la secció 2.1 que fa referència als requeriments que ha de satisfer el sistema.

Capítol 4

Test

Si observem la figura 4.1 podem veure la pantalla del sistema que ens mostra totes les converses que han estat importades prèviament. A aquesta pantalla s'hi pot accedir clicant el botó *Conversations* que apareix a la barra de navegació situada a la part superior de la pantalla. La barra de navegació, també ens mostra amb quin usuari ens hem identificat i ens permet tancar la sessió, i en cas que no estiguem identificats, també ens permet entrar al sistema proporcionant el nostre usuari i la clau de pas. Si ens fixem amb les converses que ens mostra el sistema, podem veure que tenen un color de fons. Aquest color de fons que es mostra a cada conversa, ve determinat en funció del sentiment que te associat el tweet que s'està mostrant. Per tant, el color verd indicarà un sentiment positiu, el color blanc un sentiment neutral i el color vermell un sentiment negatiu. Per a cada conversa es pot escollir mostrar tots els tweets que la formen (botó *Show*) o bé generar el graf que la representa (botó *Show graph*).

Si escollim la opció de mostrar la conversa, tal com es pot veure a la figura 4.2, podem observar que es mostren tots els tweets que formen la conversa i que el color de cada tweet ve determinat en funció del sentiment que té associat, seguint el mateix criteri que s'ha explicat anteriorment. Cada tweet conté un menú desplegable on es pot modificar el sentiment que té associat. D'aquesta manera podem corregir el sentiment dels tweets que no estiguin classificats correctament.


Quan estem visualitzant una conversa, també se'ns permet generar el graf d'aquesta a partir d'un botó situat a la part superior de la pantalla, concretament al tweet arrel. Si seleccionem aquesta opció se'ns generarà el graf que representa la conversa, tal com es pot veure a la figura 4.3. El graf que se'ns mostra segueix les especificacions que s'han explicat en seccions anteriors. En aquest cas, tal com es pot observar en la part superior de la pantalla, s'ha escollit una llegenda que coloreji els nodes de la conversa en funció del sentiment que tenen associat. Si analitzem la conversa, podem veure que hi ha alguns nodes que tenen sentiment neutral. Si es volgués determinar si els usuaris recolzen al tweet arrel, potser s'hauria d'editar la conversa i associar un sentiment negatiu o positiu als tweets que tenen un sentiment neutre. D'aquesta manera ens seria més fàcil analitzar aquest aspecte.

Si una vegada estem visualitzant el graf decidim utilitzar un altre criteri a l'hora de colorejar els nodes, podem canviar el criteri a utilitzar a partir del menú desplegable que conté el requadre que

ens mostra la llegenda. Si seleccionem un altre criteri se'ns actualitzarà el graf en funció del criteri escollit. Així doncs, si sobre el graf que teníem anteriorment canviem el criteri per què en comptes de considerar el sentiment de cada node consideri el nombre de favorits que té, ens quedaria el graf tal i com es pot veure a la figura 4.4. En aquest cas, podem veure que el tweet arrel és el node que ha estat marcat més vegades com a favorit, concretament 165. Seguidament tindriem el node 4 que ha estat desat com a favorit entre 6 i 12 vegades, i per últim la resta de nodes, que només han estat marcats com a favorits de 0 a 5 vegades. Analitzant aquestes dades arribem a la conclusió que el node 4 és una resposta rellevant, ja que en comparació amb la resta de nodes, ha estat marcat més vegades com a favorit.


TwiGraph Conversations Signed in as **sanchezfauste** [Logout](#)

Imported conversations




Postureig de Lleida @PostureigLleida
#scupft2 Sóc Top influencier a #Lleida amb un 80 de Klout, comparteixo/ajudo a causes solidàries i no cobro per això... Hi ha gent per a tot

[Show](#) [Show graph](#)




Europa Press @europapress
#lomásleído Miles de hologramas de todo el mundo 'protestan' contra la 'ley Mordaza'
<http://t.co/E5ofE16iOA> <http://t.co/nNqlvJxnJN>

[Show](#) [Show graph](#)



Roger @RogerFreeman_
El meu cosí es gradua avui a Iowa i li posen una bandera espanyola i ell diu que res de res, que l'estelada. <https://t.co/B1xOee2KJR>

[Show](#) [Show graph](#)




Policía Nacional @policia
Nuestros compañeros han evitado que una mujer se quitara la vida saltando de una azotea en #Alicante
BRAVOOO 🇪🇸🇪🇸 <https://t.co/1vpTiihwP7>

[Show](#) [Show graph](#)

Figura 4.1: Pantalla on es mostren les converses importades al sistema

TwiGraph Conversations Signed in as **sanchezfauste** Logout



Policía Nacional @policia
Importado el: 4 de Julio de 2016 a las 22:03

✓ Actualizar antivirus y SO ✓ Hacer #backup ✓ Usar links seguros ✓ Comprar en web fiables
#MaravillosaRutina que #Evitadisgustos


132 171

Neutral Guardar Show graph




David dybit @DavidInarejos
@policia que es eso del #backup ?

Neutral Guardar



Magus 1121 @Magus1121
@DavidInarejos @policia copia de seguridad de tus archivos, para si se pierden poder recuperarlos.

A favor Guardar



David dybit @DavidInarejos
@Magus1121 @policia gracias mil compañero. Saludos

A favor Guardar

Figura 4.2: Pantalla on es mostra una conversa

TwiGraph Conversations Signed in as **sanchezfauste** [Logout](#)

Sentiment ▾

Negative Neutral Positive

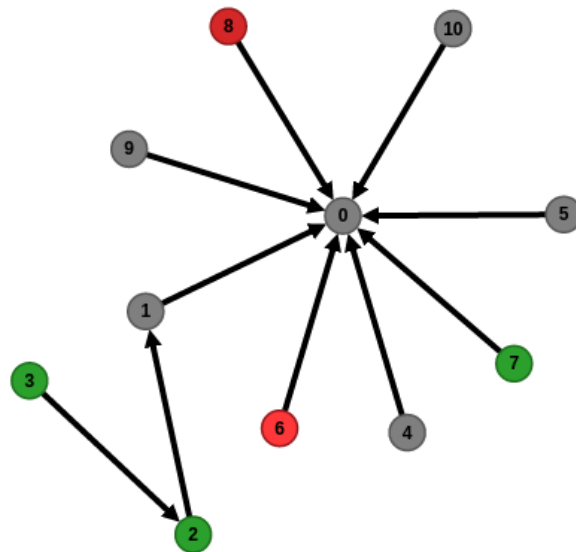


Figura 4.3: Conversa de la figura 4.2 en forma de graf - llegendes sentiment



Figura 4.4: Conversa de la figura 4.2 en forma de graf - llegendes favorits

Capítol 5

Conclusions

El projecte m'ha permès adquirir i posar en pràctica nous coneixements sobre el disseny d'aplicacions web. Aquests coneixements em resultaran molt útils a l'hora de dissenyar i posar en funcionament un servei o aplicació web.

Representar una conversa en forma de graf et permet veure quins són els tweets més importants que la formen, ja que no sempre el tweet més rellevant d'una conversa és el tweet arrel. Es pot donar el cas que el tweet que ha creat polèmica sigui una de les respostes al tweet arrel, o per exemple, que d'una conversa formada per 70 tweets n'hi hagi 5 que tenen una rellevància molt superior en comparació amb la resta.

El sistema es pot ampliar fàcilment per a què suporti altres plataformes a part de Twitter, només s'ha d'afegir un nou cas d'ús que permeti importar una conversa de la plataforma que es vulgui utilitzar. La part del sistema que s'encarrega de recuperar una conversa de la base de dades i mostrar-la a l'usuari, ja sigui de forma directa o en forma de graf, seguiria sent compatible sempre i quant és respecti el format en que es desen les converses al sistema actualment.

Es poden afegir nous criteris a utilitzar a l'hora de colorejar els nodes del graf, com per exemple, en funció del nombre de seguidors que té l'autor del tweet.

Una funcionalitat que jo crec que podria ser interessant si es segueix ampliant el projecte, seria que a partir del graf que genera el sistema hi hagi algun mecanisme que sigui capaç de detectar de forma automàtica quins són els tweets més rellevants d'una conversa. Per a dur a terme aquest procediment s'hauria d'estudiar i determinar quins factors indiquen que un tweet és rellevant (número de respostes que conté, número de retweets, etc).

Durant la experimentació amb el sistema i després d'estar analitzant diverses converses, m'he adonat que el nombre de retweets i de favorits que té un tweet no acostuma a ser un bon factor per determinar-ne la seva importància. Hi ha moltes converses que l'únic tweet que té favorits i retweets és el node arrel, i per tant, aquests atributs poden no ajudar-nos a l'hora d'esbrinar quines respostes són més rellevants dins d'una conversa.

Crec que aquest és un camp d'investigació que pot arribar a ser molt útil per saber que opina la població sobre un tema concret. Una aplicació que podria ser útil, seria determinar en temps real quin tant per cent de la població està a favor sobre algun aspecte concret que s'estigui debatent a les xarxes socials.

Bibliografia

- [1] Teresa Alsinet, Josep Argelich, Ramón Béjar, Cèsar Fernández, Carles Mateu i Jordi Planes. Weighted Argumentation for Analysis of Discussions in Twitter.
- [2] Teresa Alsinet, Josep Argelich, Ramón Béjar, Jordi Planes i Marc Sánchez. Analysis of Discussions in Twitter with an Argumentation Tool.
- [3] Twitter Inc. Documentació de l'API de Twitter. <https://dev.twitter.com/overview/api>. [Consultat el 5 de juliol de 2016].
- [4] Bootstrap. Documentació de Bootstrap. <http://getbootstrap.com/components/>. [Consultat el 5 de juliol de 2016].
- [5] Django Software Foundation. Documentació Django. <https://docs.djangoproject.com/en/1.9/>. [Consultat el 5 de juliol de 2016].
- [6] d3js. Documentació D3. <https://github.com/d3/d3/wiki>. [Consultat el 5 de juliol de 2016].
- [7] Beautiful Soup. Documentació Beautiful Soup. <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>. [Consultat el 5 de juliol de 2016].
- [8] Wikipedia. Web scraping. https://es.wikipedia.org/wiki/Web_scraping. [Consultat el 5 de juliol de 2016].
- [9] Wikipedia. El llenguatge de programació Python. <https://ca.wikipedia.org/wiki/Python>. [Consultat el 5 de juliol de 2016].
- [10] Wikipedia. Twitter Bootstrap. https://es.wikipedia.org/wiki/Twitter_Bootstrap. [Consultat el 5 de juliol de 2016].
- [11] Wikipedia. Django (framework). [https://es.wikipedia.org/wiki/Django_\(framework\)](https://es.wikipedia.org/wiki/Django_(framework)). [Consultat el 5 de juliol de 2016].
- [12] Wikipedia. JavaScript. <https://es.wikipedia.org/wiki/JavaScript>. [Consultat el 5 de juliol de 2016].
- [13] w3schools. JavaScript and HTML DOM Reference. <http://www.w3schools.com/jsref/>. [Consultat el 5 de juliol de 2016].
- [14] Mozilla Foundation. JavaScript reference. <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference>. [Consultat el 5 de juliol de 2016].