Universitat Autònoma de Barcelona

escola d'enginyeria

3677-1:
Desenvolupament d'un Memetracker - Rastrejador de noticies

Memòria del Projecte Fi de Carrera
d'Enginyeria en Informàtica
realitzat per
Xavier Rabadán Rius
i dirigit per
Ramon Grau Sala
Bellaterra, 31de Gener de 2011

El sotasignat, Ramon Grau Sala
Professor de l'Escola Tècnica Superior d'Enginyeria de la UAB

CERTIFICA:

Que el treball a què correspon aquesta memòria ha estat realitzat sota la seva direcció per en Xavier Rabadán Rius

I per tal que consti firma la present.

Signat: Ramon Grau Sala
Bellaterra, 31 de Gener de 2011

Agraïments

Voldria aprofitar aquesta oportunitat per agrair als meus pares tot el suport i paciència que m'han donat al llarg de la carrera, els consells oferts i la gran ajuda que m'ha suposat tenir-los presents durant tota la formació acadèmica.

Agrair també la col·laboració i la confiança dipositada en el projecte pel meu director de projecte, Ramon Grau Sala. Sense ell, aquest projecte no hagués estat possible.

A tots els meus companys i als grans amics que he fet durant aquests anys, ja que amb ells he passat molts dels millors moments de la meva vida.

També voldria donar les gràcies molt especialment a May Pérez i Dani Abril, per la seva ajuda incondicional, els seus consells i la paciència que han tingut en tot moment.

Per últim agraeixo a totes aquelles persones que han cregut i donat suport a aquest projecte.

Índex general

Índex de figures

Índex de taules

Capítol 1

Introducció i Plantejament

Aquest és el document de memòria pel projecte de final de carrera d'Enginyeria Informàtica Superior. En ell trobarem varies seccions on es detalla des de la planificació i anàlisi fins la fase d'implementació i posada en marxa.

1.1. Identificació, plantejament i justificació de l'originalitat del projecte

L'augment exponencial de la informació disponible en format digital durant els últims anys i les expectatives de creixement futur fan necessària l'organització de tot aquest contingut, amb la finalitat de millorar la cerca i accés a la informació. Amb aquesta finalitat, adquireix importància la relació i classificació de textos, i més específicament de notícies.

Actualment existeixen diverses pàgines web que ens permeten conèixer les notícies i els continguts més destacats de la "blogosfera", els quals apliquen algorismes similars per relacionar notícies que tracten sobre un mateix tema. Gran part d'aquestes webs en funcionament són de parla anglesa i alguns d'ells tracten sobre una única temàtica.

A continuació exposaré els diferents espais webs existents que són relacionadors de notícies:

- Espais webs en llengua anglesa:

- *www.techmeme.com*: L'exposo en primera posició ja que és el pioner i probablement és el relacionador de notícies més important i més consultat en l'actualitat tot i que tracta únicament de notícies relacionades amb la tecnologia.
- *www.blogrunner.com*: Aquest és propietat de NYTimes, el qual disposa de totes les temàtiques que es publiquen en el famós diari.
- *megite.com*: Aquest espai tracta diverses temàtiques com és Tecnologia, Política, Oci...
- *digg.com/news*: En aquesta web també podem trobar que es tracten noticies de temàtiques diferents i que ens permet de seleccionar depenent del que volem consultar en cada moment.

- *slashdot.org*: En aquesta web, com succeeix amb techmeme, només trobem notícies relacionades amb assumptes tecnològics.

- Espais webs en llengua castellana:

- *www.xgil.com*: Aquesta és una web que recopila noticies de multitud de blocs que publiquen notícies en llengua castellana de tecnologia.
- *www.tecnomeme.com*: Aquesta web fins fa ben poc funcionava de la mateixa manera que l'anterior, relacionant notícies únicament sobre tecnologia.
- *www.wikio.es*: Aquest és un dels rastrejadors més coneguts al nostre país. Rastreja i publica notícies de diferents temàtiques i les mostra en una mateixa portada.
- *www.blodico.com*: Aquest espai té diverses temàtiques, tot i que les relacions de notícies que utlitza no són del tot encertades

- Espais webs relacionadors de microblogging:

- *www.tweetmeme.com*: Aquest és un espai web que rastreja multitud de microblogs del Twitter i crea una portada principal amb els tweets més destacats.
- *www.buzrr.com*: Aquesta és una de les novetats de Google d'aquest any, fent d'aquesta pàgina web un recopilador d'allò que més s'està compartint a Google Buzz.

Segons l'estudi previ que s'ha fet de l'estat de l'art del projecte podem dir que si que existeixen entorns similars que proporcionen menys funcionalitats i que algunes d'elles no estan del tot ben implementades. El disseny de funcionalitats afegides amb intenció d'ampliació ha permès que l'usuari tingui una major interactivitat amb l'aplicació i major comoditat per així rebre el que cada usuari vulgui conèixer.

L'entorn del projecte està dissenyat de manera que es pugui fer un manteniment o ampliació de manera fàcil i còmode ja que s'ha utilitzat una programació ben estructurada. Els diferents algoritmes implementats es poden reutilitzar per altres projectes, com és la classificació de textos per idiomes i per temàtica.

Després de realitzar aquest projecte es tindrà un major coneixement sobre els diferents llenguatges i sobre les diferents arquitectures utilitzades.

1.2. Introducció

En aquesta memòria s'explica com implementar un *memetracker* per a que assoleixi els meus objectius inicials.

1.2.1. Què és un *Memetracker*?

Un *mem* (de l'anglès *meme* i aquest de *memory*) és, segons les modernes teories sobre la transmissió de la cultura a les noves generacions, la unitat mínima de transmissió de la informació.

Un *memetracker*, resumidament, el que realitza és una selecció i jerarquització de noticies publicades en les fonts seleccionades (blocs) mitjançant l'aplicació automatitzada de determinats algoritmes.

Per tant podem definir que un *memetracker* és un software per a l'estudi de la migració dels *memes* a través d'un grup de persones. El terme es fa servir normalment per a descriure les pàgines web que:

- Analitzen blocs per determinar quines pàgines web estan sent discutides o citades amb més freqüència, o
- Que permeten als usuaris votar els enllaços a pàgines web que troben interessants.

La introducció dels *memetrackers* va ser una eina fonamental en el creixement dels blocs com a forts competidors de la informació envers els mètodes tradicionals. A través de l'automatització es va fer possible trobar les millors fonts d'informació dins dels nombrosos blocs.

El factor determinant que identifica a cada *memetracker* és l'origen de les seves fonts, escollides amb un cert criteri més o menys selectiu.

1.3. Objectius

L'objectiu principal d'aquest projecte és desenvolupar un *memetracker* que ofereixi un millor servei a l'usuari d'una manera més eficient al que actualment podem trobar a la xarxa. Desenvolupar un sistema totalment controlable i que pugui ser portable a diferents temàtiques.

Per a dur a terme aquest projecte es necessitarà crear un sistema totalment automàtic que reculli i classifiqui les notícies de diferents fonts d'informació (principalment blocs de qualitat i prestigi). Aquest sistema analitzarà totes les fonts d'informació i n'extraurà i relacionarà les notícies que parlen sobre un mateix tema. A més a més el sistema etiquetarà automàticament les notícies per a una millor recerca i organització de les mateixes.

Donat que s'haurà de mostrar tota aquesta informació, s'ha decidit crear una aplicació web, amb l'afegit de les avantatges d'una xarxa social, és a dir, part dedicada als usuaris perquè puguin escollir les seves preferències i la mateixa aplicació sigui variable segons l'usuari que l'utilitzi. També es dotarà d'un arxiu amb totes les notícies que s'hagin anat publicant. L'administrador tindrà accés a una zona privada, per a poder organitzar i corregir de forma adequada qualsevol error que s'hagi pogut produir durant el funcionament de l'aplicació.

1.4. Planificació del projecte

La planificació d'un projecte ha de començar amb una estimació. Segons Frederick Brooks:
"Les nostres tècniques d'estimació estan pobrament desenvolupades. I el que és pitjor, inherentment reflecteixen una superposició que es bastant falsa, la de que tot anirà be. Com no estem segurs de les nostres estimacions, els gestors del software sovint no són capaços de convèncer a la gent de que poden esperar un bon producte" [Brooks, 1975].

Per que el grau d'incertesa de la estimació sigui menor en un projecte, s'han d'utilitzar dades de projectes anteriors que permetin ajustar els models d'estimació. Per la realització d'aquest projecte em vaig posar en contacte amb *www.blodico.com* i *www.xgil.com* i vaig esbrinar que aquestes dues webs es van desenvolupar en un any cadascuna amb un equip de tres persones

per la primera i dues per la segona, tot i que en el segon cas el disseny va ser contractat exteriorment..

Donat que l'estimació és la base de tota la resta d'activitats de planificació del projecte i que serveix per una bona enginyeria del software, no és en absolut aconsellable embarcar-se sense ella.

1.4.1. Elements d'informació bàsics que calen per desenvolupar el projecte

Els elements d'informació necessàris per desenvolupar el projecte són:

- Coneixements d'anàlisi i desenvolupament d'aplicacions PHP.
- Coneixement d'utilització d'eines de desenvolupament de software.
- Tècniques per l'emmagatzemament persistent d'informació.
- Tècniques de disseny i implementació de webs..
- Tècniques de processament de llenguatges.
- Coneixement sobre usabilitat, disseny i implementació d'algoritmes de classificació.

1.4.2. Fixar les fonts d'informació a utilitzar

Pel projecte s'utilitzen com fonts d'informació:

1. Persones:
    - Director del projecte..
    - Professors de la Universitat d'Intel·ligència Artificial

2. Altres medis:
    - Documentació sobre llibreries de PHP
    - Articles tècnics sobre alguns aspectes concrets de programació en PHP, mySQL, llenguatge M (MatLab) i C++.
    - Llibres i manuals sobre anàlisi de software, patrons de software i eines de desenvolupament.
    - Fòrums sobre programació.

1.4.3. Planificació temporal del projecte

Primerament mostro la planificació que vaig fer inicialment:

Període	Tasques a desenvolupar
Desembre '09	- Recerca d'informació, documentació i investigació d'eines similars. - Posar-me en contacte amb els autors dels blocs que formin la base de dades.
Gener '10	- Fer una recerca exhaustiva de totes les característiques del software que s'utilitzarà, i procedir a la instal·lació d'aquest.
Febrer '10	- Dissenyar la base de dades. - Treballar amb l'eina que m'ajudarà a fer la lectura *feeds* que hagi de classificar posteriorment.
Març '10	- Dissenyar i implementar l'algorisme que relacioni les notícies que parlen sobre un mateix tema.
Abril '10	- Treballar amb la part de xarxa social. - Creació del disseny de l'aplicació. - Inici del sistema relacional de notícies per poder testejar-lo i corregir-lo.
Juny '10	- Treballar amb la part de xarxa social - Creació d'un gestor intern per treballar més fàcilment amb totes les dades. - Creació del disseny de l'aplicació.
Juliol '10	- Posar en funcionament l'algorisme per diferents temàtiques. - Creació del sistema automàtic de creació de diferents estadístiques.
Agost '10	- Recopilar la informació utilitzada durant el desenvolupament del projecte i de la documentació generada i redactat de la memòria que reuneixi la feina i resultats del projecte.
Setembre '10	- Presentació del projecte

Després d'haver desenvolupat el projecte, han hagut alguns canvis en les tasques realitzades segons la planificació inicial. Mostro llavors la planificació real que he seguit durant tot el desenvolupament del projecte:

Període	Tasques a desenvolupar
Desembre '09 – Abril '10	- Recerca d'informació, documentació i investigació d'eines similars. - Posar-me en contacte amb els autors dels blocs que formin la base de dades.
Maig '10	- Fer una recerca exhaustiva de totes les característiques del software que s'utilitzarà, i procedir a la instal·lació d'aquest.
Juny '10 – Agost '10	- Dissenyar la base de dades. - Treballar amb l'eina que m'ajudarà a fer la lectura *feeds* que hagi de classificar posteriorment.
Setembre '10	- Dissenyar i implementar l'algorisme que relacioni les notícies que parlen sobre un mateix tema.
Octubre '10	- Treballar amb la part de xarxa social. - Creació del disseny de l'aplicació. - Inici del sistema relacional de notícies per poder testejar-lo i corregir-lo.
Novembre '10	- Treballar amb la part de xarxa social - Creació d'un gestor intern per treballar més fàcilment amb totes les dades. - Creació del disseny de l'aplicació.
Desembre '10	- Posar en funcionament l'algorisme per diferents temàtiques. - Creació del sistema automàtic de creació de diferents estadístiques.
Gener '11	- Recopilar la informació utilitzada durant el desenvolupament del projecte i de la documentació generada i redactat de la memòria que reuneixi la feina i resultats del projecte.
Febrer '11	- Presentació del projecte

1.5. Estructura de la memòria

En aquest document es pretén posar de manifest què i com s'ha dut a terme una solució adient per a un problema plantejat.

El documents s'estructura en cinc grans blocs: introducció i plantejament, anàlisi de requeriments, metodologia i resolució, proves i experimentació i conclusions. En el primer bloc s'explica què i com produeixen la informació les fonts de les quals disposarà el sistema. Al següent apartat s'explica tot el material que s'ha necessitat per la creació del projecte. A l'apartat de metodologia i resolució s'explica detalladament i pas per pas com s'han desenvolupat totes les parts que formen el projecte. A continuació s'expliquen les proves realitzades, per tal de testejar el sistema i les seves parts i s'analitzen tots aquests resultats. I per últim s'exposen les conclusions resultants de tota la feina realitzada en aquest projecte, juntament amb les idees de treball futur per tal de millorar l'aplicació.

Capítol 2

Anàlisi de requeriments

En aquest capítol es tracten alguns conceptes necessaris pel desenvolupament del projecte.

2.1. Materials i eines

En aquest apartat es descriuen els programes i llibreries utilitzats al projecte.

EINA	UTILITAT
Wamp	- Provar en local la base de dades - Provar en local el codi implementat - Provar de manera offline l'aplicació
PhpEdit	- Desenvolupament de l'aplicació - Desenvolupament dels diferents plugins integrats en l'aplicació
CakePHP	- Desenvolupament del control d'accés a la part privada per part dels usuaris.
PhpMyAdmin	- Creació de la base de dades en MySQL - Administració de la base de dades en MySQL
Matlab	- Desenvolupament de l'algoritme de referenciació de notícies - Desenvolupament de l'algoritme de classificació per idioma - Desenvolupament de l'algoritme de classificació per temàtica
SimplePie	- Extracció de dades d'una notícia en XML - Parsejar els diferents estàndards de redifusió de contingut
OSU SVM	- Desenvolupament de l'algoritme de referenciació de notícies

Taula 2.1. Programes i llibreries utilitzades en el projecte

2.1.1. Wamp

A més a més del servidor contractat s'ha utilitzat el software *Wamp* (la seva versió per Windows), un paquet de programari lliure.

El terme WAMP és un acrònim per descriure la plataforma sobre la que funcionen les aplicacions web creades utilitzant la següent combinació d'eines:

- *Windows*, el sistema operatiu;
- *Apache*, el servidor web
- *MySQL*, el servidor de bases de dades;
- Perl, PHP, o Python, llenguatges de programació.

2.1.2. PHPEdit

Aquesta ha sigut l'eina principal pel desenvolupament de l'aplicació web, ja que al ser un software potent i especialitzat (sobretot en llenguatge PHP) ha facilitat i agilitzat molt la feina.

Algunes de les eines que incorpora són la il·luminació i el completament automàtic de les comandes, un debugger intern per detectar errors en el codi i un suport per treballar amb més d'un document simultani.

2.1.3. CakePHP

És un marc de desenvolupament (*framework*) ràpid per PHP, lliure i de codi obert. Es tracta d'una estructura que serveix de base als programadors per que aquests puguin creaer aplicacions Web. L'objectiu principal és que es pugui treballar de manera estructurada i ràpida, sense perdre flexibilitat.

2.1.4. PHPMyAdmin

PHPMyAdmin, una eina escrita en llenguatge PHP per a la creació i administració de *MySQL* a través de pàgines web, utilitzant un navegador. Gràcies a aquesta eina la implementació de la base de dades ha sigut un procés fàcil i sobretot àgil.

Actualment pot crear i eliminar Bases de Dades, crear, eliminar i alterar taules, borrar, editar i afegir camps, executar qualsevol sentencia SQL, administrar claus en camps, administrar privilegis, exportar dades en diversos formats i està disponible en 50 idiomes. Es troba disponible sota la llicencia GPL.

2.1.5. Matlab

MATLAB (abreviatura de MATrix LABoratory, “laboratori de matrius”) és un software matemàtic que ofereix un entorn de desenvolupament integrat amb un llenguatge de programació propi (llenguatge M). Permet manipular fàcilment matrius, dibuixar funcions i dades, implementar algorismes, crear interfícies d'usuari, i comunicar-se amb altres programes en altres llenguatges.

Aquesta eina ha sigut clau pel desenvolupament de l’algorisme relacional i dels algoritmes de classificació.

2.1.6. SimplePie

SimplePie és una llibreria per PHP que ens permet parsejar d’una manera fàcil i ràpida els diferents estàndards de redifusió de contingut, RSS(versions 0.9 ,1.0 i 2.0) i Atom.

Existeixen diferents llibreries amb les mateixes funcionalitat, però aquesta és una dels més populars i actual, i a més a més ofereix un entorn totalment configurable i modular.

2.1.7. OSU SVM

OSU SVM és una *toolbox* gratuïta per *Matlab* que incorpora una Màquina de Suport Vectorial (Support Vector Machine). Cal remarcar que està programada amb llenguatge C++ i això fa que sigui una eina molt ràpida.+

Aquesta llibreria m'ha permès de desenvolupar els algoritmes clau per la referenciació de les notícies.

2.2. Conceptes previs pel desenvolupament del projecte

En aquest apartat es descriu quins coneixements he d'adquirir per poder desenvolupar el projecte.

2.2.1. Fonts d'informació

Cada cop és més freqüent trobar pàgines web d'informació sobre l'actualitat diària. Alguns dels exemples més clars de generadors d'informació són els diaris digitals, que des de fa uns anys posen a disposició dels usuaris la seva versió electrònica. També podem trobar agències de notícies o fins i tot podem parlar dels blocs, que ara estan tan de moda.

Avui dia tota pàgina informativa té activat el servei de redifusió de contingut web, anomenat canal web, el qual s'explicarà detalladament en el següent punt.

Totes les fonts d'informació utilitzades en el desenvolupament d'aquest projecte estan mencionades en l'annex A1.

2.2.2. Canal web

Un canal web és un fitxer XML que conté una llista de les últimes notícies publicades per la pàgina en format RSS o Atom. Aquest servei de redifusió de la informació permet compartir-la de una forma ràpida i senzilla.

El gran avantatge d'aquests canals web és que utilitzant el metallenguatge XML tenen una gramàtica específica definida, per tant l'anàlisi i l'extracció de la informació és extremadament senzilla.

2.2.3. Utilització d'arxius XML

Cada notícia està definida per l'etiqueta *item*, i dintre d'aquesta podem veure les diferents etiquetes que defineixen totes les parts de les quals està definida aquesta.

Les etiquetes que defineixen les parts de cada notícia són les següents:

- *Title*: títol de la notícia.
- *Link*: direcció web on es troba aquesta notícia.
- *Guid*: identificador únic de la notícia. (camp opcional)
- *Pubdate / Published*: defineix la data de publicació del ítem. (camp opcional)
- *Author / Dc:Creator*: autor de la notícia. (camp opcional)
- *Description / Content:Encoded*: cos de la notícia.
- *Category*: element que especifica una o més etiquetes per a definir el ítem. (camp opcional)

Per obtenir les notícies de la xarxa recuperem les direccions dels arxius de publicació XML, de cada una de les fonts d'informació que prèviament han estat introduïes a la bases de dades per un administrador.

Per tal d'analitzar cada un dels arxius XML es fa ús d'una llibreria de programari lliure anomenada *SimplePie*. Aquesta s'encarregarà de transformar cada un dels ítems de cada arxiu en un vector, amb el qual podrem accedir molt fàcilment a totes les parts que composen aquell ítem per a poder-lo emmagatzemar posteriorment a la base de dades.

2.3. Requeriments funcionals

En aquest apartat analitzaré quins són els requeriments funcionals que ha de tenir l'aplicació per a que sigui competitiva, atractiva i original.

2.3.1. Recol·lecció i anàlisi relacional

- Recollir les últimes notícies de totes les fonts d'informació.
- Fer un anàlisi exhaustiu de les notícies per poder-les relacionar amb les altres segons les seves característiques.

2.3.2. Usuari web de la zona pública de l'aplicació

Els requeriments funcionals per la zona pública, on qualsevol usuari pot accedir, són els següents:

- Veure els diferents grups de notícies relacionades de la portada (les més recent publicades), on per cada grup es podrà apreciar:
    - Una petita informació associada a la primera notícia que ha parlat sobre aquell tema (títol, autor, bloc, data i la introducció).
    - La resta de títols i la seva procedència de les demés notícies que composen el grup.
    - Temàtica de la que tracta la notícia.
- Veure els títols de les últimes notícies recollides, estiguin o no relacionades.
- Veure els títols de les notícies més citades de la setmana.
- L'usuari podrà ampliar la informació de cada notícia seleccionant el títol d'aquesta, ja que estarà enllaçat amb la pagina on ha estat publicada la notícia.
- Capacitat per poder comunicar-se amb l'administrador de l'aplicació.

2.3.3. Gestor de l'aplicació

Funcionalitat d'accés restringit, on només hi podrà accedir l'administrador. En aquesta aplicació haurà de permetre controlar i modificar totes les accions que prèviament s'hagin fet automàticament pels algoritmes desenvolupats, és a dir els que recullen, relacionen i classifiquen les notícies.

2.3.3.1. Descomposició funcional detallada del gestor d'aplicació

Aquestes eines són les següents:

- Llistar/Afegir/Modificar/Eliminar les temàtiques que es desitja que formin part de l'aplicació (ex: tecnologia, esports, salut,...)
- Llistar/Afegir/Modificar/Eliminar les diferents fonts d'on s'extraurà la informació.
- Modificar/Eliminar tan les notícies com el seus diferents elements, objectes de codi (vídeos, àudios, etc.)
- Llistar/Afegir/Modificar/Eliminar les diferents etiquetes.
- Llistar/Afegir/Modificar/Eliminar relacions entre les notícies.
- Llistar/Afegir/Modificar/Eliminar paraules per poder-les afegir en llistes negres, blanques o fins i tot de sinònims, per tenir un major control de les paraules que aporten o no informació.
- Llistar/Afegir/Modificar/Eliminar enllaços web en llistes negres, per així, poder-los ignorar.

Capítol 3

Metodologia i resolució

En aquest apartat es desenvolupa la proposta que es fa del projecte que es vol portar a cap.

3.1. Arquitectura de l'aplicació

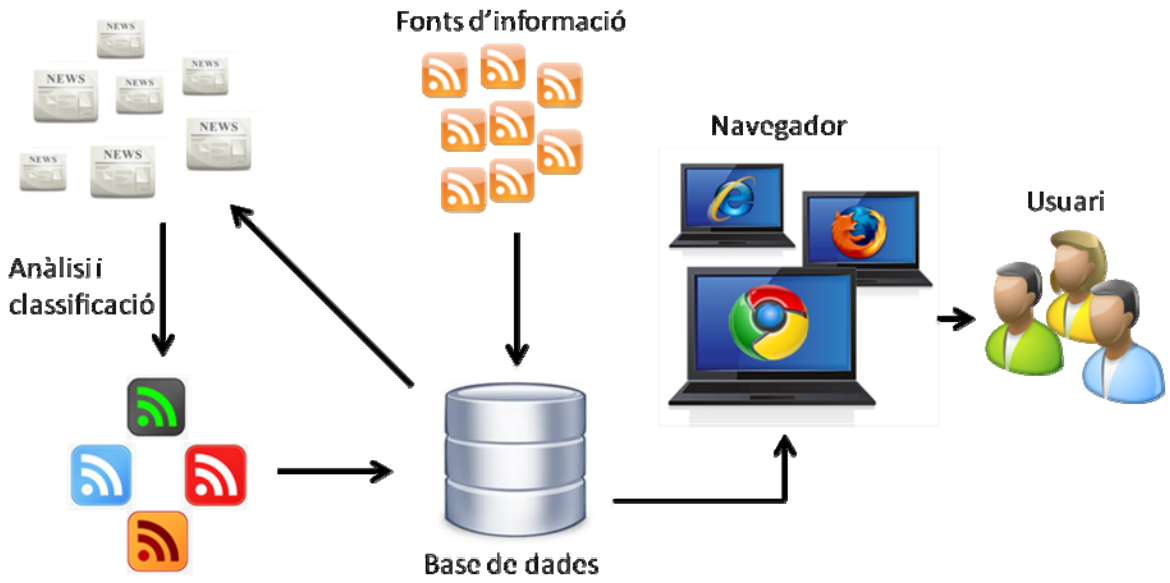

Figura 3.1. Esquema de funcionament del projecte

3.2. Requeriments del sistema

En aquest apartat es tractarà el disseny i els aspectes rellevants de la implementació en profunditat.

Els requeriments de ELTEUBLOG es poden classificar en requeriments generals i requeriments d'usabilitat.

3.2.1. Requeriments generals de l'entorn

És possible que alguns d'aquest requeriments entrin en conflicte entre sí:

- Compatibilitat amb diferents sistemes operatius i navegadors web, provat en els diferents navegadors i sistemes operatius actuals.
- Ha d'executar-se amb fluïdesa i ha de ser totalment automàtic, sense que la seva funcionalitat depengui de l'administrador per ser executat.
- La interfície ha de complir els requeriments d'usabilitat establerts.
- Connexió a Internet, tant per mostrar resultats com per obtenir nova informació.
- Sistema robust a atacs de SQL injection.
- Software fàcil d'usar, mantenir i ampliar.

3.2.2. Requeriments d'usabilitat de l'entorn

Es busca que l'usuari experimentat es manegi ràpida i intuïtivament per l'aplicació i que un nou usuari tingui les mínimes dificultats possibles quan comenci a utilitzar-la.

- Totes les funcionalitats de l'aplicació has de ser executades mitjançant el ratolí o el teclat.
- La interfície s'executarà en una mateixa finestra del navegador, és a dir que no s'obriran noves finestres o pestanyes en cap moment.
- Els missatges d'informació que es mostren, utilitzaran text el més directe i resumit possible per augmentar la probabilitat de que l'usuari el llegeixi i l'entengui ràpidament.
- Les opcions s'agruparan correctament per categories.
- Llegibilitat. El text de la interfície s'ha de poder llegir amb facilitat, ha de ser suficientment gran i ha d'haver contrast entre el color del text i els del fons. Les combinacions de colors seran harmonioses.

3.3. Modelat i disseny del projecte

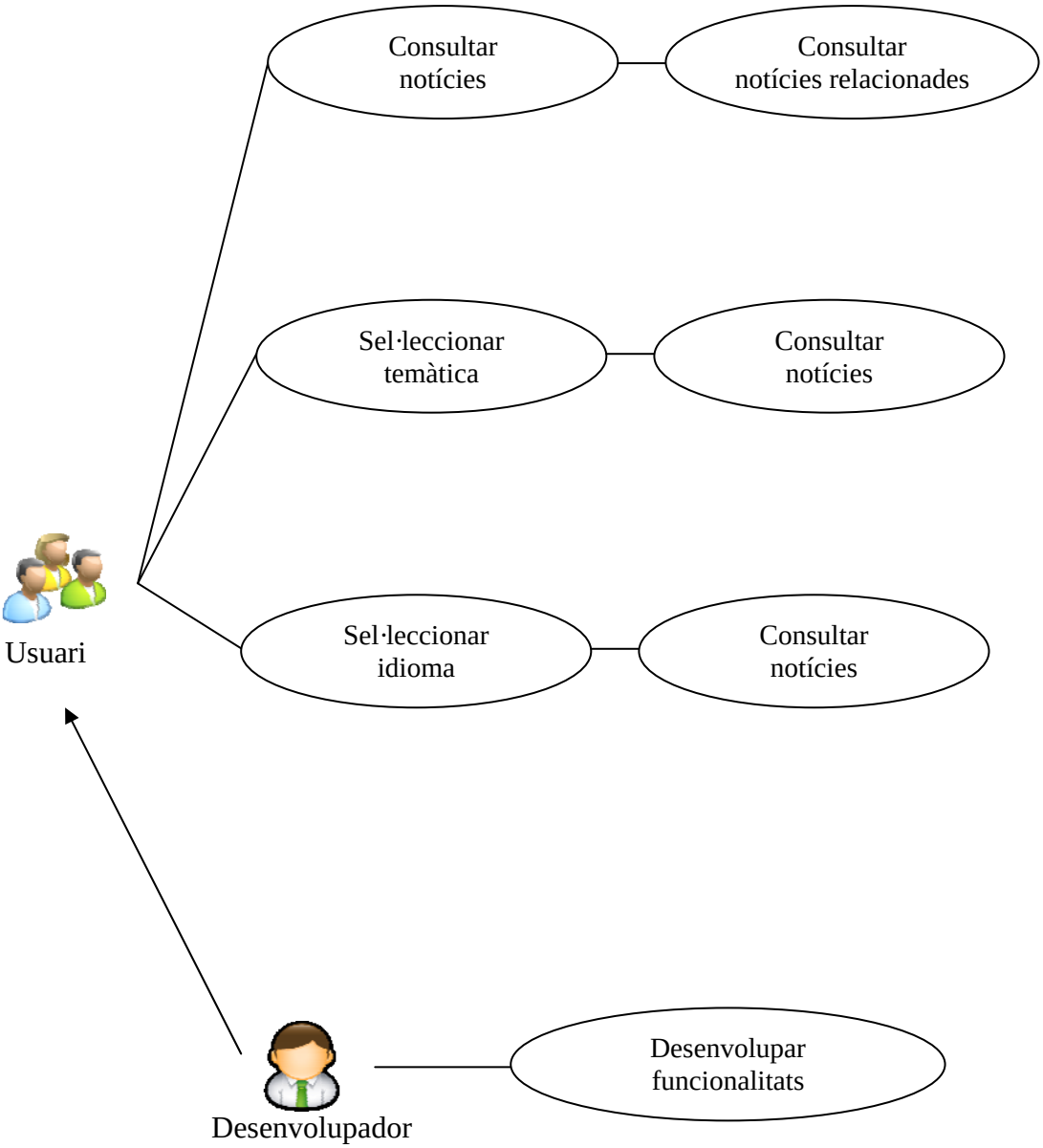

Figura 3.2. Diagrama de casos del projecte

3.4. Disseny i implementació de l'entorn del projecte

En aquest apartat es mostren les solucions als subproblemes en els que es descompon el projecte. Es presenta cada subproblema, s'exposa i justifica la solució presa.

3.4.1. Base de dades

La base de dades és un dels pilars bàsics, ja que pateix consultes i modificacions constantment per tots els processos que formen el projecte.

A continuació s'expliquen totes les taules de la base de dades creada per poder gestionar, controlar i consultar tot el flux de dades que té l'aplicació:

3.4.1.1. Notícies

Els flux de notícies està controlat principalment per tres taules:

- **Item_queue:** Taula on es guardaran totes les dades de cada notícia recollida, com per exemple el títol, cos, autor, un indicador de l'existència d'objectes *html* (video, audio, etc.)... i apart de totes les dades representatives de la notícia guardem l'identificador *processed* per saber si la notícia ha estat processada, és a dir, si ha passat pel procés d'anàlisi i classificació.

- **Item_current** i **Item_archive:** Aquestes dues taules tenen una igual estructura i a l'hora també la mateixa que l'anterior, excepte que aquestes no contenen el camp *processed*, és a dir, l'indicador per saber si la notícia ha sigut tractada. Aquestes són omplertes amb les notícies que s'han emparellat, deixant a la taula *item_queue* les notícies que no s'han processat o que no han pogut ser relacionades amb altres.

  Com s'ha comentat no hi ha cap diferencia estructural entre aquestes dues taules, però tenen una funcionalitat molt diferent: la primera taula ens servirà per emmagatzemar notícies que encara poden ser relacionades, mentre que la segona emmagatzemarà totes les notícies que s'han relacionat.

La decisió de crear aquestes dues taules i tenir duplicats alguns registres ha estat envers al número elevat de consultes que es fan a l'hora de relacionar les notícies i al número tant alt de registres que es poden arribar a emmagatzemar amb el temps. Per tant, totes les consultes necessàries es faran sobre la taula *item_current*, la qual només contindrà les notícies que es poden relacionar. Mentrestant, la taula *item_archive* contindrà totes les notícies que s'hagin relacionat, es puguin o no seguir relacionant i només es farà servir pel mostratge i consulta de notícies per part dels usuaris.

Després d'haver explicat les tres taules principals de la base de dades s'expliquen les taules que ens permeten emmagatzemar i controlar algunes de les característiques de les dades.

- **Url:** Emmagatzema les diferents direccions electròniques que conté cada notícia en el seu cos. Cal comentar que les direccions de cada notícia només es guardaran a la base de dades mentre la notícia encara pugui ser relacionada amb altres.

- **Vídeo:** Emmagatzema els diferents objectes *html* (àudios, vídeos, etc.) que conté cada notícia en el seu cos.

- **Groups:** Taula encarregada de guardar parelles de relacions entre notícies. Aquesta consta de dos camps; el primer, *id_Pitem*, és l'identificador de la notícia declarada com a pare i *id_item* és l'identificador de la notícia relacionada amb l'anterior.

3.4.1.2. Fonts d'informació

En aquest apartat s'expliquen aquelles taules que ens permeten gestionar tot el relacionat amb les fonts d'informació d'on extraiem les notícies que tractarem.

- **Subject:** Taula que contindrà les diferents temàtiques de les fonts d'informació, segons hagi assignat el classificador de temàtica, de les quals disposarà l'aplicació, juntament amb el seu identificador.

- **Web_feed_source:** Aquí es trobarà tota la informació relacionada amb les fonts d'informació, com per exemple la direcció del seu fitxer *XML*, el nom, la direcció web, una petita descripció, direcció de la imatge, el seu nom en versió curta i per últim

un camp que ens indica l'estat actiu o passiu segons estiguem tenint o no en compte aquesta font per a obtenir informació.

3.4.1.3. Gestió

En aquest apartat s'expliquen les taules que utilitzem per els algoritmes que formen .

- **Word:** Paraules útils per l'administració. Aquestes estaran presents en les taules que expliquem a continuació.

- **Black_list:** Relació entre una paraula de la taula *word* i una temàtica de la taula *subject*, per així crear una llista de paraules negres, classificades per les diferents temàtiques.

- **Synonymous:** Relació entre dues o més paraules de la taula *word*, per així poder definir sinònims.

- **Black_list_url:** Taula que permet emmagatzemar diferents enllaços juntament amb una temàtica, per així crear una llista d'enllaços negres, classificats per les diferents temàtiques.

3.4.2. Sistema de recol·lecció

L'aplicació es divideix en diversos subproblemes, el primer del qual és el sistema de recol·lecció de notícies. Per aquesta part s'ha desenvolupat un sistema de recollida i emmagatzemament automàtic de notícies, les quals en un procés posterior seran analitzades i classificades.

Abans d'emmagatzemar les notícies cal fer un tractament de les dades. El primer que farem serà separar les diferents *metadates* que puguin tenir les notícies, ja que seran útils en el procés de relació. Realitzant aquest procés d'extracció de *metadates* aconseguim per una banda estandaritzar les dades, ja que depenent de la font d'informació poden tenir diferents codificacions (les més habituals són *iso-8859* o *UTF-8*) i, per altra banda, volem evitar el problema del *SQL Injection*, tot i que no s'esperen atacs d'aquest tipus, si que es poden

produir errors al introduir les dades a la base de dades, utilitzant el mateix concepte de l'atac, és a dir, que les dades continguin certs caràcters com, per exemple, les cometes simples.

Les notícies que volem recollir s'han d'extreure dels canals web de cada una de les fonts d'informació. Per tant, el primer pas a fer és recuperar de la xarxa les direccions dels arxius de publicació XML que prèviament han estat introduïdes a la base de dades per un administrador.

Per analitzat cada un dels arxius XML es fa ús d'una llibreria de programari lliure anomenada *Simple Pie* i que em permetrà transformar cada un dels ítems de cada arxiu en un vector per així poder accedir fàcilment a totes les parts que composen aquell ítem per poder-lo emmagatzemar posteriorment a la base de dades.

3.4.3. Sistema de classificació per idioma

Aquest apartat tracta sobre com es processen les notícies que anem obtenint i emmagatzemant per etiquetar-les segons el llenguatge de text que utilitzen.

Cal a dir que després de documentar-me i analitzar les diferents opcions per solucionar el problema de classificar textos segons el llenguatge que estigui escrit he optat per aplicar un algorisme que durant la carrera havíem utilitzat. Aquest algorisme és el *Naive Bayes* que en els següents apartats explicaré.

3.4.3.1. Introducció a Naive Bayes

Els classificadors de la família de *Naive Bayes* són molt utilitzats en la tasca de classificació de textos degut a que produeixen resultats comparables amb els obtinguts per altres mètodes més sofisticats i són relativament senzills d'implementar.

El classificador de *Naive Bayes Simple* considera la probabilitat d'aparició de cada terme donada la classe de forma binaria, és a dir el terme apareix o no i llavors la seva probabilitat condicional donada la classe és o no considerada. En aquest sentit, el classificador *Naive Bayes Multinomial* sol millorar l'exercici doncs considera el número d'aparicions del terme

per avaluar la contribució de la probabilitat condicional donada la classe amb el que el modelat de cada document s'ajusta millor a la classe a la que pertany.

Pot pensar-se que si es modifica la representació dels documents, de manera que el recompte dels termes que apareixen en ell es canvia pel número d'aparicions del terme a la classe la probabilitat de pertinença del document del qual s'està avaluant, s'està proporcionant informació addicional al classificador per a que l'assignació de classe millori.

3.4.3.2. Classificador de textos

La tasca de classificació de textos consisteix en assignar a cada document d'una col·lecció una etiqueta que designa a la classe a la que pertany aquest document. La decisió sobre quina etiqueta ha de ser assignada a un document determinat es pren a partir d'un model construït com una part de la col·lecció denominada conjunt d'entrenament. L'objectiu de la tasca és construir un model que predigui correctament la classe d'un conjunt de documents.

Quan es desitja millorar la classificació per un domini donat es solen utilitzar dos maneres diferents:

- Modificar el model del classificador: ja sigui canviant-lo per un altre o alternats els seus paràmetres per adaptar-lo a les dades que ha de classificar. Existeix una gran quantitat de classificadors utilitzats en aquesta tasca com *Arbres de Decisió, Màquines de Vectors de Suport, k-NN* o models probabilístics entre els que es troben les *Xarxes Bayesianes* i *Naive Bayes*.

- Modificar la representació de les dades: diferents classificadors produeixen millors o pitjors models segons la representació de les dades que es desitja processar, l'elecció d'una representació adequada donat un classificador és un problema per sí mateix. Hi ha varies maneres de representar un document abans de processar-lo en un classificador però entre les més utilitzades destaca el model vectorial en el que cada document és representat com un vector de dimensió igual a la mida del vocabulari de la col·lecció i en el que el valor de cada atribut correspon al recompte d'aparicions del terme corresponent en el document, tot i que també es sol utilitzar una representació

binaria (1 si el terme apareix en el document sense importar el número d'aparicions, 0 si no és així) o altres esquemes de pesat.

Per solucionar aquest problema es desenvolupa un model probabilista, particularment *Naive Bayes Multinomial* per classificar documents d'una col·lecció de notícies amb una representació vectorial i un esquema de pesat que considera la freqüència d'aparició de cada bi-grama possible en la llengua per l'entrenament del model. Posteriorment la representació dels documents de proba es modifica utilitzant un esquema de pesat per classes.

3.4.3.3. Classificador Naive Bayes Multinomial

El model probabilístic *Naive Bayes* és un dels més simples i més utilitzats en classificació de textos perquè produeix resultats tan bons com altres models més sofisticats.

Es basa en l'aplicació de la *Regla de Bayes* per predir la probabilitat condicional de que un document pertanyi a una classe $P(c_i \mid d_j)$ a partir de la probabilitat dels documents donada la classe $P(d_j \mid c_i)$ i la probabilitat a priori de la classe en el conjunt d'entrenament $P(c_i)$

$$P(c_i \mid d_j) = \frac{P(c_i)P(d_j \mid c_i)}{P(d_j)}$$

Donat que la probabilitat de cada document $P(d_j)$ no aporta informació per la classificació, el terme es sol ometre. La probabilitat d'un document donada la classe sol ser assumida com la probabilitat conjunta dels termes que apareixen en aquests documents donada la classe i es calculen com:

$$P(d_j \mid c_i) = \prod_{t=1}^{|V|} P(\omega_i \mid c_i)$$

Addicionalment, el model *Naive Bayes Multinomial* considera la freqüència d'aparició de cada terme en els documents x_i enlloc d'una ocurrència binaria:

$$P(d_j \mid c_i) = \prod_{t=1}^{|V|} P(\omega_i \mid c_i)^{x_t}$$

El terme $P(w_t \mid c_i)$ es calcula a partir del número d'aparicions de cada terme w_i en una classe c_i però per evitar el problema de les probabilitats 0 s'utilitza l'estimació de *Laplace*

$$P(w_t \mid c_i) = \frac{1 + n(w_t, c_i)}{|V| + n(c_i)}$$

On $n(w_t, c_i)$ és el número d'ocurrències de w_t a c_i , *vert*$|V|$ és la mida del vocabulari i $n(c_i)$ és el recompte total de paraules a c_i. D'aquesta manera, la classificació es fa buscant l'argument que maximitza la funció:

$$c^*(d) = \arg\max_{ci}\ p(c_i) \prod \prod_{t=1}^{|V|} P(\omega_i \mid c_i)^{x_{tt}}$$

3.4.3.4. Dades

En aquest procediment s'utilitzen els llibres de la trilogia del *Senyor del anells* en castellà i en anglès per entrenar i provar el sistema. La trilogia d'aquest títol és idònia per realitzar l'aprenentatge de l'algorisme ja que són textos actuals i molt extensos, amb una mitja de 210000 paraules per llibre. Per tant estem fent un entrenament per idioma amb més de 600000 paraules.

Per poder tractar les paraules que apareixen en els textos s'han passat els llibres de format *pdf* a *txt*. Un cop acabat aquest procés el que farem serà fer una neteja que consisteix en eliminar tots els símbols, utilitzant funcions pròpies de *PHP* que permeten eliminar aquests símbols mitjançant la utilització d'expressions regulars, de tal manera que només quedin lletres.

3.4.3.5. Muntatge pràctic

Entrenament del model

La manera d'aconseguir fer l'entrenament el primer que farem serà separar en bigrames els textos, tenint en compte també els espais, és a dir que contarem per exemple

‘ab’, ‘bc’, etc. Per realitzar aquest procediment construïm un algorisme en *Matlab* que tractarà cada text com si fos un vector i anirà recorrent-lo.

El resultat que ens dona l’algorisme és una matriu de dimensió 39 x 39, per a cada un dels llenguatges que tractem (castellà i anglès), a on el valor contingut a cada parell de coordenades *(i,j)* correspon al nombre de vegades que es troba el bigrama *ij* en el text introduït. El valor de cada atribut correspon al recompte d’aparicions d’aquest terme al document. Per exemple, el bigrama *ab* correspon a la posició *(2,3)*. Si en aquesta posició de la matriu hi ha trobem un valor de *15*, vol dir que el bigrama mencionat es repeteix 15 vegades en el text.

Com ja s’ha comentat anteriorment l’algorisme que utilitzem no té en compte els signes d’interrogació i tampoc tenim en compte el doble espai en blanc com a bigrama, però si l’espai en blanc seguit de caràcter i els caràcters seguits d’espai en blanc, per tenir en compte amb quina lletra comencen o acaben les paraules en cada un dels idiomes.

En les figures 3.3 i 3.4 es mostra el recompte per un dels textos utilitzats en castellà.

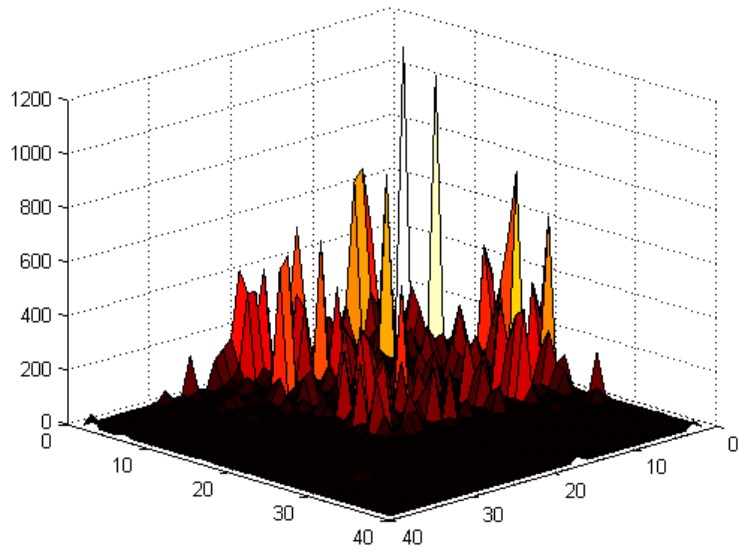

Figura 3.3. Mostra en 3D del recompte de bigrames del text en castellà

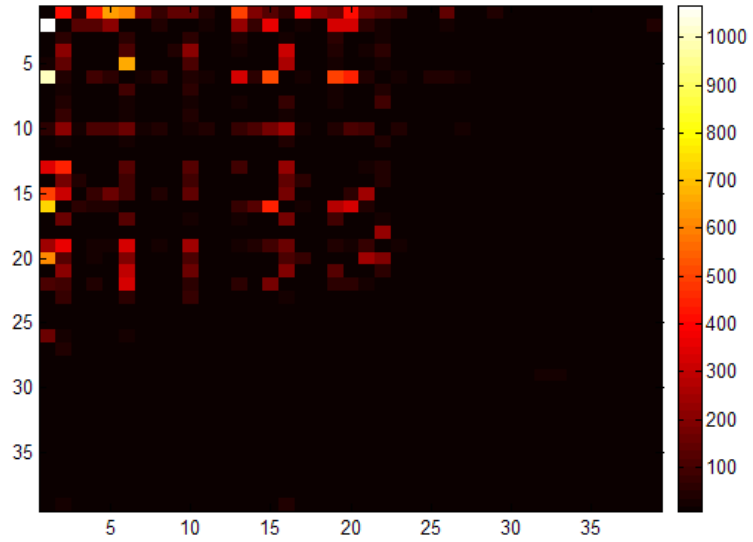

Figura 3.4. Mostra en 2D del recompte de bigrames del text en castellà

En el text tractat en castellà veiem que els atributs que dominen estan en les coordenades *(2,1), (6,1)* que correspon als bigrames *'a_'* i *'e_'* respectivament, és a dir que moltes paraules acaben amb caràcter *'a'* i amb el caràcter *'e'*.

En les figures 3.5 i 3.6 es mostra el recompte per un dels textos utilitzats en anglès.

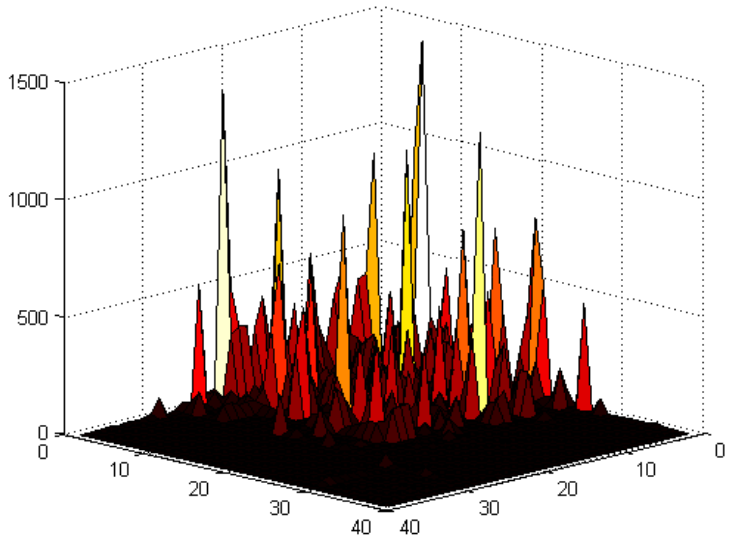

Figura 3.5. Mostra en 3D del recompte de bigrames del text en anglès

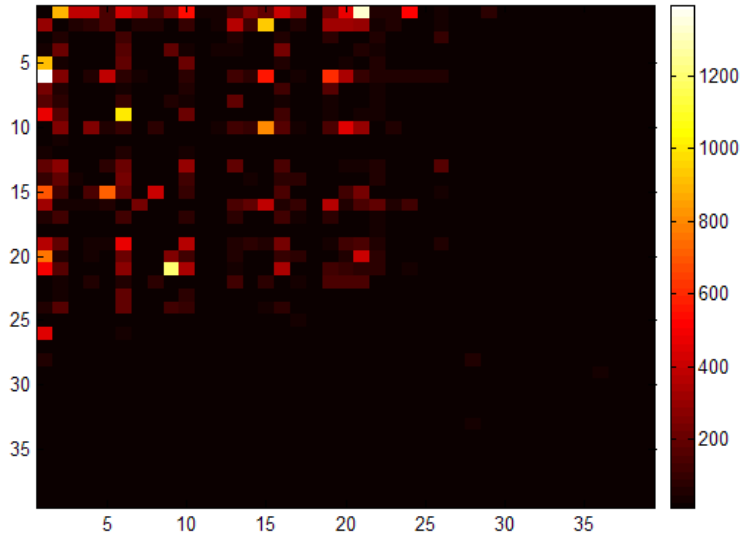

Figura 3.6. Mostra en 2D del recompte de bigrames del text en anglès

En aquest cas els bigrames que dominen sobre els altres són *'e_'*, *'_t'*, *'th'* i *'he'*. Aquests bigrames són clarament diferents als dels casos anteriors.

Per mostrar aquests resultats gràficament hem utilitzat les comandes *surf* de *Matlab* per visualitzar els valors en una superfície tridimensional, i la comanda *imagesc* que ens mostra un pla amb punts en una escala de colors que ens permet veure en quines coordenades de la matriu hi ha els valors més alts.

En el cas de les gràfiques en 3D s'ha rotat la figura fins a deixar l'origen de coordenades al fons de la imatge de manera que els valors no obstrueixin la visió dels altres valors més petits. Notar que els valors de l'eix Z en el cas de les gràfiques en 3D estan representats en milers.

Un cop fet el recompte de bigrames dels tres textos tornem a fer un algorisme que construeixi una matriu de probabilitats que serà el model que s'utilitzarà en la classificació.

Prova del model

El conjunt de prova que utilitzem són notícies extretes d'un bloc d'història de cada un dels dos idiomes que posteriorment hem etiquetat per poder comprovar els resultats. El conjunt de prova també és traslladat a una representació vectorial amb els mateixos atributs que el conjunt d'entrenament i es prova el model segons l'equació característica del classificador *Naive Bayes Multinomial* avaluant, la probabilitat de pertinença a cada una de les classes.

Els atributs amb major probabilitat s'escull com la classe de cada un dels documents de prova i posteriorment es compara amb l'etiqueta de classe per avaluar l'exercici del classificador.

3.4.4. Sistema de classificació per temàtica

En aquest apartat s'explica la manera que s'ha utilitzat per classificar les notícies entre les tres temàtiques escollides: cuina, fotografia i motor.

3.4.4.1. Introducció i objectius

L'augment exponencial de la informació disponible en format digital durant els últims anys i les expectatives de creixement futur fan necessària l'organització de tot aquest contingut, amb la finalitat de millorar la cerca i accés a la informació. Referent a aquest objectiu, adquireix importància la investigació de la classificació automàtica de textos.

La classificació automàtica de textos es basava, en els seus inicis, en els seus inicis, en tècniques d'enginyeria del coneixement, on un expert definia de manera manual les regles que cada document tenia que complir per pertànyer a una o altra categoria. No obstant, el gran cost que suposava això, junt amb els avanços que s'havien realitzat en l'àrea de la intel·ligència artificial, va donar lloc als anys 80 a la utilització de tècniques d'aprenentatge automàtic per aquests propòsits. Des de llavors, han estat molts els mètodes utilitzats per la classificació automàtica de textos.

L'aprenentatge automàtic tracta d'obtenir, de manera automatitzada, les característiques que ha de complir un objecte per ser classificat en una determinada categoria, basant-se en una

col·lecció inicial de documents preclassificats. Així, una vegada que s'han obtingut els descriptors per cada classe, el sistema els utilitza per crear un classificador, podent classificar nous documents.

Actualment la majoria de les tècniques per la construcció de classificadors automàtics es basen en aprenentatge automàtic. Aquest aprenentatge pot ser de tres tipus diferents, segons la seva base de coneixement:

- **Aprenentatge supervisat:** a partir d'una col·lecció d'entrenament s'aprenen les característiques que ha de complir un document per pertànyer a una o altra classe, creant posteriorment el classificador. Un cop acabada aquesta fase d'entrenament, el classificador final està definit i s'utilitza per la categorització de documents dels que no es coneix la seva classe.

- **Aprenentatge semisupervisat:** la fase de creació del classificador utilitza la col·lecció d'entrenament com a base, però es segueix refinant amb documents sense classificar. En aquests casos, el número de documents sense classificar sol ser molt més gran que el dels ja classificats. Aquest tipus d'aprenentatge fa que la disposició d'un número reduït de documents preclassificats no sigui un problema per la classificació automàtica, i s'eviti el treball costós de tenir que etiquetar o aconseguir una gran col·lecció preetiquetada, però generalment es més crítica la creació d'un bon classificador.

- **Aprenentatge no supervisat:** en aquests casos s'extreuen els patrons de classificació sense la disposició d'una col·lecció de documents preclassificats. La classificació és, per tant una agrupació en grups sense etiquetar, el que s'anomena *clustering*.

Les tècniques d'aprenentatge automàtic utilitzades inicialment eren en la seva majoria supervisades, el que suposa la disposició de grans col·leccions de documents prèviament etquetats per la fase d'entrenament.

En els últims anys, les màquines de vectors de suport (Support Vector Machines, SVM), en vista dels bons resultats que ofereixen, es perfilen com una bona solució pels problemes de classificació automàtica.

Tenint en compte tot això, s'ha considerat la tasca de classificació de textos com un problema multiclasse, on generalment es disposa d'una taxonomia de més de dues categories, i en la que la utilització de documents etiquetats podria resultar útil.

3.4.4.2. Classificació automàtica de textos

En general, es coneix com classificació automàtica a la tasca d'assignar una o varies categories predefinides sobre una col·lecció d'instàncies a classificar. Per fer això, s'assigna un valor booleà per cada parell (d_j, c_i), on d_j és qualsevol document de la col·lecció de documents $D = d_1, \ldots, d_n$, i c_i qualsevol categoria del conjunt de categories predefinides $C = c_1, \ldots, c_k$.

La tasca de la classificació automàtica sol estar composta, generalment, de les següents subtasques:

- **Representació:** Els documents de text o les instàncies que composen la col·lecció de dades a classificar ha de ser transformat comprensiblement pel sistema de classificació que es vagi a utilitzar. Per fer això, s'han d'identificar les característiques representatives pels documents, amb la finalitat de que la representació sigui adequada.

- **Classificador:** La subtasca de classificació pot dividir-se, a la vegada, en dues fases: entrenament i test. A la fase d'entrenament s'alimenta el sistema de classificació amb els documents que ja han passat la fase de representació, amb la finalitat d'extreure els descriptors de classe. Un cop realitzat això, la fase de test s'ocupa de predir les categories corresponents pels documents per classificar.

- **Avaluació:** Finalment, a l'haver predit les categories corresponents als documents per classificar, es procedeix a avaluar els resultats, per comprovar la seva qualitat.

3.4.4.3. Experiments

En aquesta secció es presenten els motius que han portat a la decisió d'utilitzar *SVM* com tècnica de classificació, al marge d'altres aproximacions.

3.4.4.3.1. Perquè SVM?

Les tècniques basades en *SVM* estan donant lloc al desenvolupament de múltiples estudis sobre classificació de textos. Així mateix, degut al gran interès que ha creat, son moltes les implementacions que es poden trobar de les variants d'aquest algorisme.

L'algorisme *SVM* es basa en un model espai vectorial (VSM) igual que moltes altres tècniques d'aprenentatge, però aporta algunes avantatges sobre les altres.

- No es requereix una selecció o reducció de termes. En el cas de que una classe es distribueixi en àrees separades de l'espai vectorial, serà la redimensió mitjançant la funció *kernel* la que s'ocupi de solucionar-ho.

- No és necessari realitzar un esforç d'ajustament de paràmetres en el cas de problemes linealment separables, ja que disposa del seu propi mètode per fer-ho.

No obstant, la naturalesa pròpia de *SVM* es limita a la classificació binaria supervisada, el que fa interessant l'estudi de portar-ho més enllà, tal i com s'explica a continuació.

3.4.4.3.2. Problemàtica de SVM

L'algorisme *SVM* s'ha convertit en una de les tècniques més utilitzades per fer la classificació automàtica, degut als bons resultats que s'han obtingut. Aquesta tècnica es basa en la representació dels documents en un model espai vectorial, i assumeix que els documents de cada classe són separables en l'espai de representació: en base a això, tracta de buscar un hiperplà que separi ambdues classes. Entre tots els hiperplans que separen les classes, *SVM* es queda amb aquella que maximitza la distància entre els documents de cada classe i el propi hiperplà, el que s'anomena marge.

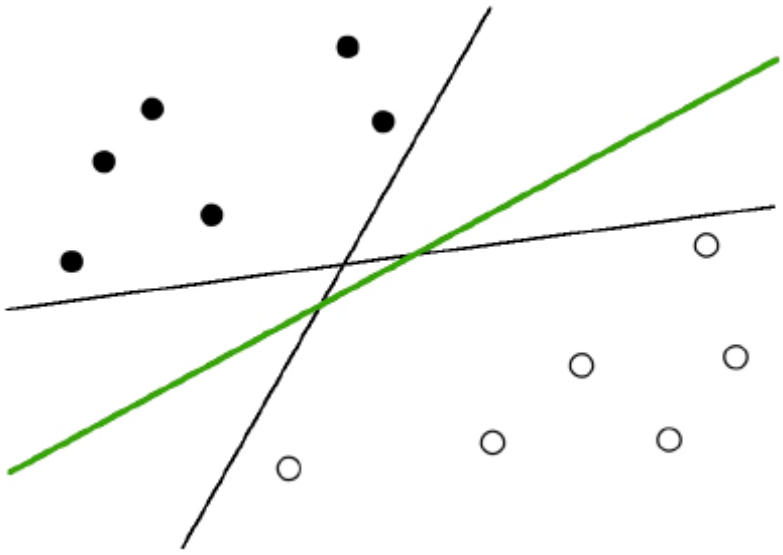

Figura 3.7. Selecció de l'hiperpla que maximitza el marge amb *SVM*

Aquest hiperplà es defineix mitjançant la següent funció:

$$f(x) = \omega \cdot x + b$$

SVM multiclasse

Degut a la naturalesa dicotòmica de *SVM*, va sorgir la necessitat d'implementar nous mètodes que poguessin resoldre problemes multiclasse. Diverses tècniques s'han basat en la combinació de classificadors binaris:

- *One-against-all* construeix *k* classificadors que defineixen *k* hiperplans que separen la classe *i* dels *k-1* restants.

- *One-against-one* construeix $\frac{k(k-1)}{2}$ classificadors, un per cada parell de classes possible, enfrentant així a totes les classes una a una.

3.4.4.3.3. Experimentació

Per la realització de la experimentació s'ha procedit a la implementació d'un algoritme *SVM multiclasse* i la seva execució sobre les col·leccions de dades escollides. Tots els documents de les col·leccions utilitzades s'han etiquetat, pel que cada una d'elles s'ha dividit en una col·lecció d'entrenament, que serveix per a que el classificador aprengui, i una altra de test, que serveix per que el sistema pugui crear les prediccions i es pugui avaluar el seu rendiment.

Les col·leccions de documents que s'han utilitzat per aquesta experimentació són un nombre de 500 notícies per cada una de les temàtiques i per cada un dels idiomes, és a dir, tenim 500 notícies en anglès i 500 en castellà etiquetades com a fotografia, 500 en anglès i 500 en castellà etiquetades amb la classe cuina i 500 en anglès i 500 en castellà etiquetades com a motor. Aquest nombre de notícies són les que tindrem en compte alhora de classificar les notícies que tractem en l'aplicació.

Des de la col·lecció d'entrenament s'han creat diferents versions, entre les que varia el número de documents etiquetats, deixant la resta com no etiquetats, podent provar així les diferents aproximacions.

Per la implementació del mètode de classificació es requereix un classificador supervisat multiclasse. S'han implementat els corresponents mètodes pel comportament del classificador supervisat i les tècniques *one-against-all i one-against-one* semisupervisades. Tots ells estan basats en aprenentatge transductiu, pel que utilitzen els documents de la col·lecció de test per seguir refinant la funció de classificació.

Un cop finalitzat l'entrenament el que ens resulten són dos algoritmes que utilitzarem alhora de fer la classificació, depenent de l'idioma que hagi resultat del sistema anterior que ens classifica el conjunt de notícies entre castellà o anglès.

3.4.5. Sistema de relació entre notícies

Aquest apartat tracta sobre com es comparen i relacionen les notícies que anem obtenint i emmagatzemant.

Després de documentar-me i analitzar les diferents opcions de comparació he decidit aplicar un algorisme de pesos a un vector de representació. El vector de representació tindrà un cert nombre de característiques i a més a més es farà un estudi mitjançant algorismes d'intel·ligència artificial per triar els millors pesos per aquests elements.

3.4.5.1. Representació de les dades

En primera instància s'han de triar totes les característiques que crec oportunes i necessàries per poder representar la unió entre dues notícies que parlen sobre un mateix tema, utilitzant els recursos de cada notícia que s'han emmagatzemat anteriorment en la base de dades. Posteriorment s'ha de trobar una manera de representació que permeti un lecturai anàlisi fàcil i senzill.

Seguidament s'expliquen quines són les característiques que s'han tingut en compte per dir si dues notícies parlen sobre un mateix tema.

A. Diferència de temps

Aquesta característica mostra els segons que resulten de la resta dels valors dels temps de publicació de les dues notícies.

Aquesta característica té una implicació rellevant ja que fa de filtre entre notícies molt semblants que estan distants en el temps. Un exemple molt clar està en notícies que tractin sobre versions de software on l'algorisme de relació les trobaria pràcticament iguals, però realment no ho son.

En la figura 3.8 es mostra com agafant un conjunt de relacions de notícies (1000 relacions) totes elles positives, les diferències de temps queden compreses en un període de 0 a 182801

segons (50,7 hores = 2.1 dies) i tenen una mitja de 47848 segons (13.49 hores). Segons aquests resultats s'ha arribat a la conclusió de triar un rang de temps de 172800 segons, és a dir 2 dies, per poder relacionar dues notícies, de no ser així ja es descartaria.

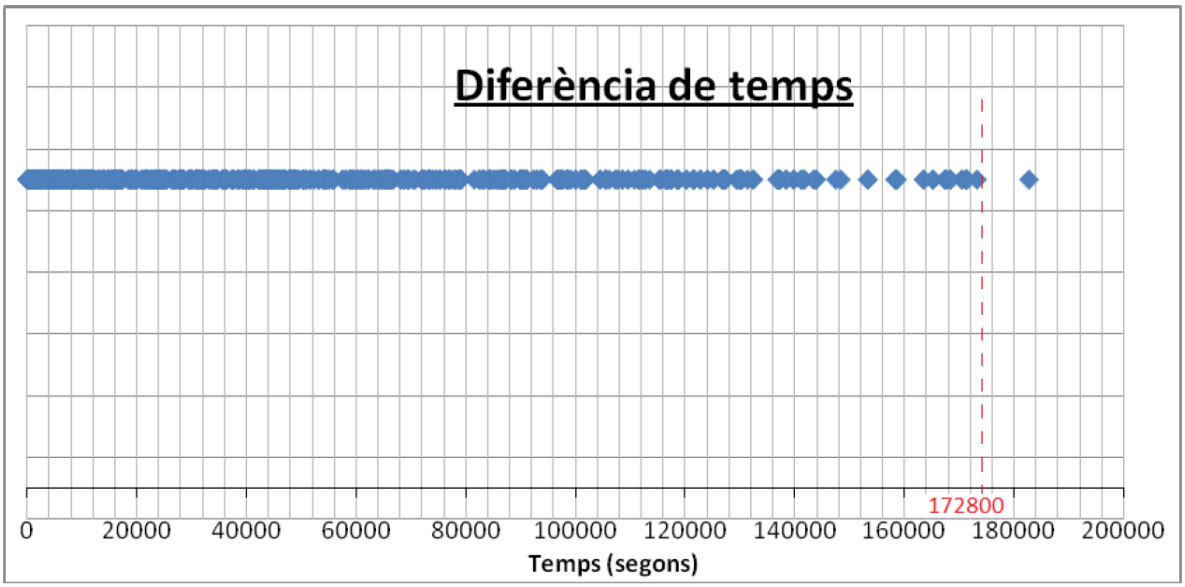

Figura 3.8. Diferència de temps entre notícies relacionades.

B. Freqüència de les paraules repetides en els títols de les notícies

La segona característica que tenim en compte és el número de paraules iguals que apareixen en els dos títols de les notícies que estem comparant i se n'obtindrà una freqüència.

Per la realització d'aquest procés hem de tenir en compte que hi ha paraules que no aporten cap contingut, com per exemple els connectors o les preposicions. Un altre problema que ens trobem és que els títols no sempre estan formats únicament per paraules, sinó que a més a més ens podem trobar símbols, com poden ser els signes de puntuació, les cometes simples o dobles,...

Per poder solucionar aquests problemes, trobem que la millor manera de fer-ho és fent una neteja prèvia al processament dels títols. El primer pas d'aquesta neteja consisteix en eliminar tots els símbols, utilitzant funcions pròpies de PHP que permeten eliminar aquests símbols mitjançant la utilització d'expressions regulars, de tal manera que només quedin lletres i números. Un cop fet aquest pas el que farem és eliminar totes les paraules que no aporten ca tipus de significat. Per eliminar aquestes paraules s'utilitza la llista negra de la base de dades,

per tant per eliminar aquestes paraules només caldrà fer unes consultes a la base de dades comparant aquestes paraules amb les de la llista negra.

Cal comentar que les paraules que formen la base de la llista negra són d'una llista de paraules buides extreta d'Internet.

A continuació es procedirà a comparar les dues noves cadenes de paraules resultants d'aquest procés, obtenint així, el número de paraules repetides que posteriorment dividirem pel número total de paraules de les dues cadenes.

Nota: Per fer les comparacions entre paraules s'utilitza una funció de PHP anomenada *similar_text()*, la qual calcula el tan per cent de similitud entre dues cadenes de text. A la taula 3.1 es poden veure alguns valors de retorn d'aquesta instrucció. Com no interessa que la similitud sigui molt gran per tal de no confondre paraules s'ha escollit considerar dues paraules iguals quan la similitud entre aquestes és d'un **84%**, d'aquesta manera podem detectar petits canvis alhora d'escriure paraules.

	anuncio	**anunció**	**anuncia**	**anunciar**	**anuncios**
anuncio	100	80	85.714	80	93.333
anunció	80	100	80	75	75
anuncia	85.714	80	100	93.333	80
anunciar	80	75	93.333	100	75
anuncios	93.333	75	80	75	100

Taula 3.1. Exemple de la instrucció *similar_text*

C. **Número de paraules dels títols que apareixen en els enllaços que conté l'altra notícia**

La tercera característica que tenim en compte és la de comparar el número de paraules del títol que apareixen també en els enllaços que conté l'altra notícia.

Un cop feta la neteja dels títols es procedeix a buscar aparicions de cada una de les paraules d'una notícia dintre dels enllaços web de l'altre. D'aquesta manera s'uneixen dos dels punts importants de les notícies, el títol i els enllaços per tal de trobar paraules claus de la notícia.

Per mostrar clarament aquest procediment poso un exemple de dues notícies relacionades. En aquestes notícies hi apareixen les paraules *'samsung', pantalla i st700*, el que farem és retornar el número de paraules iguals en el títol i en l'enllaç.

Exemple:

Títol noticia 1: Las nuevas **Samsung** de doble **pantalla**. **ST700**, PL170 y PL120

Títol optimitzat noticia 1: Samsung **doble pantalla ST700** PL170 PL120

Enllaç noticia 2: http://es.engadget.com/2011/01/05/**samsung**-dualview-**st700**-pl170-y-pl120-mas-**pantalla**-frontal-para/

D. Freqüència de les paraules repetides en els cossos de les notícies

El procés per extreure aquesta característica es similar al que s'utilitza per extreure la segona característica, però en aquest cas s'obté la freqüència de repetició de paraules dels dos cossos. Per tal d'assolir l'objectiu seguirem el mateix procediment que en el segon punt però en aquest cas no utilitzarem la funció *similar_text* entre les paraules, ja que els textos són més llargs i l'algorisme que segueix aquesta funció té una complexitat molt alta.

E. Número d'enllaços web en comú

Per aconseguir aquesta característica aprofitem que en la fase de recol·lecció de dades hem extret els enllaços del cos de la notícia. En aquest punt s'utilitzaran per comparar si dues notícies comparteixen aquestes direccions.

Els passos de comparació d'aquesta part són simples consultes a la base de dades, comparant si els enllaços d'una notícia també els conté l'altre, retornant com a valor resultant el número d'enllaços que comparteixen.

Cal comentar que cada direcció es compara dues vegades, per una banda amb les tres *www* i per l'altra sense, això es degut a que es tracta amb moltes fonts d'informació diferents i es poden escriure les direccions web de les dues maneres diferents. Com que no totes les direccions es poden escriure de les dues maneres es va descartar el fet d'estandarditzar-les en el moment de la seva extracció.

F. Indica si una notícia cita el *URI*[1] d'una altra.

Aquesta característica es basa en la comparació entre les seves direccions URI. Com sabem cada pàgina es identificada per una URI o identificador únic, això vol dir que si una notícia cita a una direcció d'un altre notícia és molt probable que estigui fent referència a la mateixa notícia. Com en el pas anterior, també és un procediment molt senzill, simplement s'ha de buscar la direcció única de la notícia dintre de les direccions web que conte l'altra. En cas de trobar-la retronarà un '1' i en cas contrari un '0'.

G. Indica si les dues fan referència al mateix objecte HTML.

Per acabar, aquesta última característica que tenim en compte és aquella que comprova els objectes HTML que apareixen en els cossos de les notícies. En el moment de l'obtenció de notícies se n'extreuen els objectes HTML i s'emmagatzemen a la base de dades. Aquests poden ser de qualsevol tipus, com per exemple vídeos, àudios, lectors de PDFs o fins i tot jocs en Flash.

Gràcies a aquesta extracció, en aquest últim punt podem comparar si dos objectes de dues notícies són iguals retornant el valor '1' si es considera que és el mateix o un '0' en cas contrari.

Per comparar dos objectes s'han aplicat dos mètodes. El primer agafa els dos objectes i els compara com si de dues cadenes de text es tractes, d'aquesta manera amb l'ajuda de la funció de PHP *str_cmp* es pot comprovar fàcilment si les dues cadenes són completament iguals.

El problema del primer mètode és que els codis poden ser diferents, ja que tots els objectes poden ser modificats per l'usuari, és a dir, es pot modificar la mida, el color o les opcions que incorporen aquests, de tal manera que, per exemple, un mateix vídeo pot estar representat per molts codis diferents.

La solució a aquest problema ha estat extreure la URI de l'objecte de dintre dels paràmetres d'aquest mitjançant la funció *preg_match*, que ens permet extreure sobre una cadena de text

[1] *URI* (identificador uniforme de recursos) és un text curt que identifica sense equivocació qualsevol recurs (servei, pàgina, document, direcció de correu electrònic, enciclopèdia, etc.) accessible en una xarxa.

una altra que compleixi el patró donat de tal manera que comparant les dues direccions obtingudes es pot saber si es tracta del mateix objecte.

Vector de característiques

Un cop hem mostrat quines són les característiques que es tenen en compte i quina és la manera d'aconseguir-les, ja podem representar-les mitjançant un vector de característiques que identifiqui una comparació entre dues notícies. En la taula 3.2 es pot apreciar la forma que tindrà aquest vector, juntament amb un exemple.

Exemple:

ID 1	ID 2	A	B	C	D	E	F	G
2598	2434	74443	0.182	1	0.118	0	0	0

Taula 3.2. Vector de característiques d'una relació entre dues notícies.

Els camps ID1 i ID2 són els identificadors únics de les dues notícies i la resta de valors són els valors que han pres les diferents característiques un cop fetes totes les comprovacions.

3.4.5.2. Obtenció de pesos

Un cop finalitzada la manera de representació de les relacions entre dos noticies procedim a recol·lectar un conjunt de dades per tal de poder fer les proves pertinents. A més a més de la recol·lecció de dades s'ha decidit fer un etiquetatge manual d'un grup de 500 relacions indicant si són positives o negatives. A la taula 3.3 es pot veure un exemple agafat de l'atzar per veure la manera d'etiquetar les relacions.

Vector de característiques	Resultat
(106, 87, 107170, 0, 0, 0.019, 0, 0, 0)	0
(106, 258, 113817, 0, 0, 0.071, 0, 0, 0)	0
(106, 86, 101873, 0.214, 1, 0.407, 1, 1, 0)	**1**
(106, 254, 99026, 0, 0, 0.0477, 0, 0, 0)	0

Taula 3.3. Etiquetatge de les relacions

Després d'etiquetar tot el conjunt de relacions, el següent pas és trobar uns pesos òptims per tal que l'algorisme de relació funcioni d'una manera eficient. Per aconseguir aquests valors es va estar buscant el millor algorisme dintre del camp de la intel·ligència artificial i es va decidir crear un algorisme genètic degut a la seva eficiència en l'optimització i la seva facilitat d'implementació.

Per representar els pesos s'utilitza un vector de set posicions associats a les diferents característiques de les relacions, de tal manera que cada pes es multiplicarà amb la seva característica. Les sis primeres posicions d'aquest vector són els pesos de les sis últimes característiques (des de la B – freqüència de paraules repetides en els títols, fins la G – referències a objectes *HTML*). En l'última posició del vector de pesos hi mostrarem el llindar que haurà de superar el producte escalar del vector de pesos amb el vector de característiques per tal de poder considerar aquella relació com a positiva.

Per implementar aquest algorisme es va decidir utilitzar *Matlab*, ja que al tractar-se d'un procès independent a l'aplicació web utilitzar *PHP* no era un requisit necessàri. A més a més *Matlab* és molt més ràpid en els càlculs, sobretot a l'hora de manipular matrius com és el nostre cas.

Els algorismes genètics funcionen creant inicialment un conjunt de solucions aleatòries conegudes com la població. En cada pas de l'optimització, es calcula la funció de cost per tota la població per tal d'obtenir una llista classificada de solucions. En aquest cas la població són els diferents vectors de pesos.

La funció de cost aplicada calcula el producte escalar entre un element de la població i un vector de característiques del conjunt calculat en l'apartat anterior. Un cop obtingut el producte escalar es compara amb l'últim element del vector de pesos per saber si existeix una relació. Aquest procés es repeteix amb tots els vectors del conjunt de característiques, i la funció cost s'encarrega de retornar l'error total de relacions que ha tingut el vector de la població amb tot el conjunt de vectors de característiques.

Seguidament a la classificació de les solucions es crea una nova població. Primerament, les millors solucions en la població actual s'afegeixen a la nova població directament i la resta de

població està formada per solucions completament noves que es generen al modificar les millors solucions. Aquest procés es pot realitzar de dues maneres:

- Mutació: Aquesta primera modificació es basa en un petit canvi aleatori a una solució existent . En aquest cas s'augmentarà o disminuirà un dels calors de la solució aleatòriament.

- Creuament: Aquest mètode implica agafar dues de les millors solucions i combinar-les d'alguna manera. Per realitzar aquesta modificació, s'agafa un número aleatori d'elements d'una solució i es completa amb la resta d'elements d'una altra solució, com es mostra en la figura 3.9.

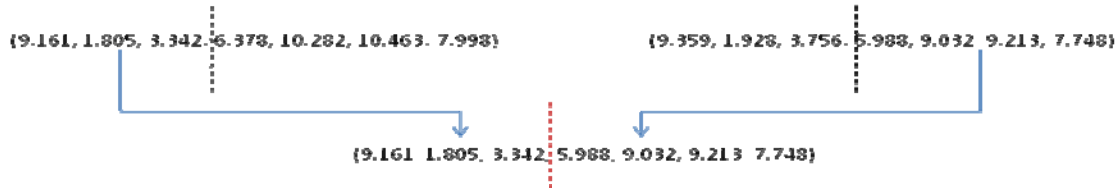

Figura 3.9. Exemple de creuament

La nova població es crea mutant aleatòriament les millors solucions. Desrpés el procès es repeteix, es classifica i es crea una altra població. Aquest procediment continua fins aconseguir un número màxim d'iteracions o fins que no s'aconsegueixen millores sobre varies generacions. En la taula 3.4 figura es poden observar alguns dels millors resultats obtinguts.

Vector de pesos	Error
(9'359, 1'928, 4'01, 5'988, 9'032, 9'213, 7'798)	0.0009
(9'609, 4'489, 1'904, 5'779, 9'404, 10'584, 7'198)	0.0009
(7'378, 1'678, 3'26, 5'988, 9'282, 8'963, 7'498)	0.0009
...	...

Taula 3.4. Exemple de resultats obtinguts

3.4.5.3. Millora de l'aprenentatge de l'algorisme

Un cop trobat un bon algorisme trobem que podríem millorar-lo, ja que l'etiquetatge del conjunt de dades s'ha fet manualment però per un conjunt petit de mostres i per tant no és tant fiable com esperem.

Per poder millorar l'aprenentatge s'ha fet ús de diverses metodologies que permeten fer un etiquetatge de les relacions més precís. El procediment serà la classificació de totes les relacions del conjunt de dades per diferents algorismes, començant per la metodologia actual i un estàndard en els mètodes de classificacions mitjançant aprenentatge supervisat, la Màquina de Suport Vectorial (*SVM*), en aquest cas utilitzem un conjunt de dades de 100000 relacions.

Un cop obtinguts els resultats de la *SVM* es comprova si de les 500 relacions classificades anteriorment hi ha cap diferència amb les classificacions obtingudes per la Màquina de Suport Vectorial, i en les relacions que obtenim un resultat diferent es procedeix a fer una nova revisió manual, obtenint així uns etiquetatges molt més fiables dels que teníem fets fins al moment.

Per implementar aquest procés s'ha creat un programa en *Matlab* que uneix les dues parts de la solució i que mitjançant la implementació creada anteriorment del algorisme trobat en aquest projecte, la *toolbox OSU-SVM*, una connexió a la base de dades i el conjunt de dades obtingut i etiquetat. Tot això ens permetrà obtenir aquesta millora en l'etiquetatge de les dades.

El procediment que es segueix per aquesta metodologia és, primerament, dividir el conjunt de dades inicial, de forma que s'aconsegueixi un conjunt per l'aprenentatge i un altre pel test. Aquesta divisió es farà utilitzant la tècnica de validació creuada *holdout*, la qual ens retornarà dos conjunts aleatoris de dades, amb la diferencia que el conjunt d'aprenentatge tindrà el 90% de les dades i el de test el 10% restant.

A continuació es farà l'aprenentatge del mètode *SVM* utilitzant el nucli lineal amb el conjunt de dades corresponent. Un cop entrenat aquest classificador es procedirà a fer la part de proves amb el segon conjunt de dades, l'extret per fer aquesta tasca.

El següent pas, un cop obtinguts els resultats del mètode anterior, és utilitzar el conjunt de test per aplicar la funció de cost juntament amb els pesos obtinguts en l'apartat anterior. D'aquesta manera ja tenim dos resultats diferents pel conjunt de test, els del algorisme *SVM* i els del propi amb pesos.

Un cop arribat a aquest punt entra en acció la segona part de la solució, que s'encarrega d'examinar els resultats obtinguts pels diferents mètodes i els compara. En el moment que es detecten diferents resultats per una mateixa relació, aquesta passa a ser examinada manualment amb l'objectiu de treure de dubtes el resultat de la relació. Com s'ha comentat, quan hi ha discrepància en els resultats d'una relació el mateix programa s'encarrega de connectar amb la base de dades, obtenir la informació de les dues notícies que formen la relació a examinar i mostrar-la per pantalla, a més a més de bloquejar-se fins que s'introdueixi per pantalla un valor ('1' o '0') indicant si les dues notícies són iguals o no.

Tot aquest procés es repetirà fins que els diferents classificadors obtinguin un 100% dels valors iguals o les relacions ja estiguin etiquetades manualment, aconseguint així una millora en els resultats de les unions de les notícies.

Seguidament en la taula 3.5 es pot veure la matriu de confusió del mètode *SVM*. Com es pot apreciar el conjunt de dades no es tan gran com el del cas anterior degut a la tècnica de validació creuada (*Holdout*), la qual crea un conjunt de test del 10% de les dades totals.

		Actuals	
		Ítems iguals	Ítems diferents
Prediccions	Ítems iguals	9	1
	Ítems diferents	26	9972
Total		35	9973
Error		0.0026	

Taula 3.5. Matriu de confusió: SVM

3.4.5.4. Optimització del vector de pesos

L'aplicació de la tècnica de l'apartat anterior ha permès millorar l'etiquetatge del conjunt de dades. Per aquest motiu ens servirà per a recalcular els valors dels pesos que utilitzarà el nostre algorisme, i així poder-los optimitzar amb els nous resultats del conjunt de dades.

De la mateixa manera que en l'apartat 3.4.2.2. obtenim els nous pesos, però tenint en compte els nous etiquetatges. A la taula 3.6 es poden veure les millores en els resultats obtinguts juntament amb el seu error produït, el qual pràcticament no ha canviat.

Vector de pesos	Error
(9'61, 3'5, 4'01, 5'988, 9'032, 9'213, 7'998)	0.0008
(9'359, 1'928, 4'01, 5'988, 9'032, 9'213, 7'998)	0.0008
...	...

Taula 3.6. Vectors de pesos després de l'optimització

3.4.6. Interfície web

Un cop realitzats els diferents passos d'anàlisi, relació i classificació de notícies, el següent pas consisteix en desenvolupar una interfície web per que permeti la publicació i consulta per part dels usuaris de les notícies tractades.

La primera part serà un gestor des del qual l'administrador podrà controlar certs aspectes del procés anterior per tal de millorar el seu funcionament. I la segona part serà la part pública en la que tots els usuaris podran veure els resultats obtinguts al aplicar els mètodes anteriors.

Cal comentar que com aquest projecte és a nivell acadèmic no he cuidat el disseny ni la presentació, és operatiu però no és atractiu.

3.4.6.1. Sistema de gestió de l'aplicació

El gestor de l'aplicació és pot comparar a un gestor de contingut d'una web autogestionable que tant s'està utilitzant últimament. Al gestor de l'aplicació s'hi pot accedir via web, però és restringida, protegida mitjançant la validació d'un usuari i contrasenya. D'aquesta protecció s'encarrega *Apache*, mitjançant l'arxiu de configuració de directoris *.htaccess* i l'arxiu *.htpasswd*, el qual conté la informació *usuari:contrasenya*.

A la figura 3.10 es pot veure la pàgina inicial d'aquest gestor, on es mostren totes les opcions que aquest ofereix a l'administrador. En els següents apartats explico totes les funcionalitats que ofereix el gestor i que separo en quatre grups.

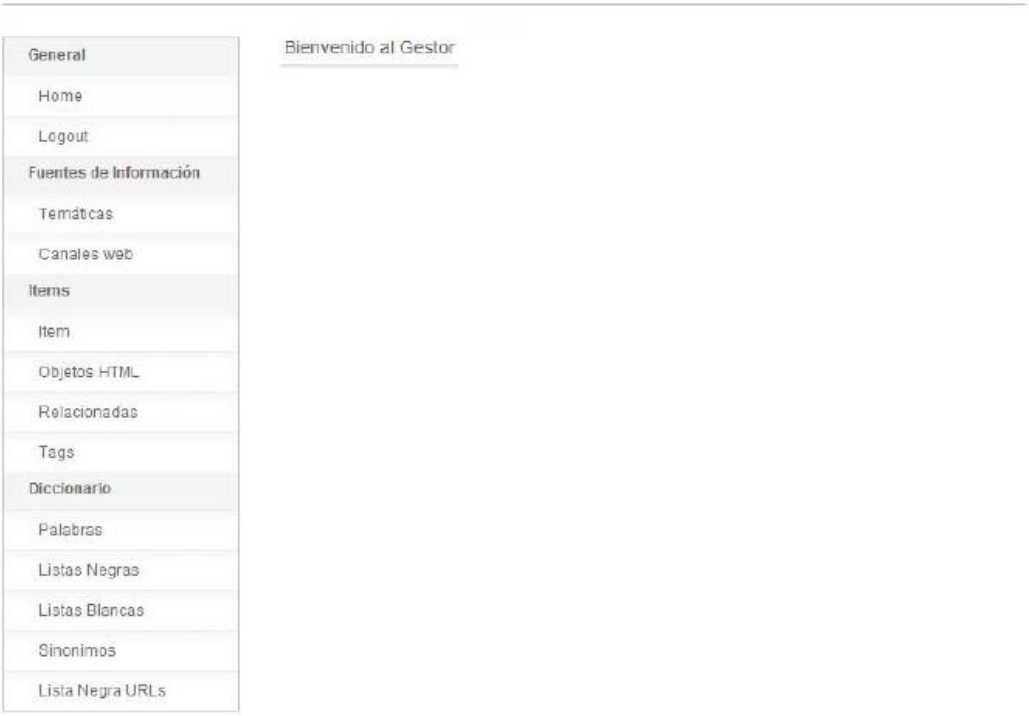

Figura 3.10. Pàgina inicial del gestor

3.4.6.1.1. Menú: General

Les primeres dues opcions estan emmarcades en el grup anomenat 'General' on trobem les opcions de *home* i logout. El primer ens retorna a la pàgina principal del gestor, la qual podem veure en la figura 3.10 anterior. La segona opció ens permet sortir de manera segura del gestor, redirigint-nos a la plana principal.

3.4.6.1.2. Menú: Fonts d'informació

Aquest grup conté també dues opcions, que són les claus per tractar les fonts d'informació i les diferents temàtiques de les notícies que tractem en l'aplicació.

A l'apartat de temàtiques s'hi mostra un formulari per afegir noves temàtiques que es volguessin tenir, tot i que per afegir una nova temàtica primer s'hauria de desenvolupar i entrenar el classificador de temàtiques per a que pugui analitzar i donar resultats tenint en compte la nova temàtica. Es mostra també una llista amb les temàtiques afegides anteriorment, a la que completem amb dues icones indicant que a les llistes se'ls hi pot modificar el nom o fins i tot eliminar-les de la base de dades.

Com a segona opció tenim l'apartat de 'Canals web' on hi podem apreciar un formulari per afegir les fonts d'informació. La introducció de fonts d'informació és extremadament senzilla; simplement hem d'afegir la direcció de l'arxiu XML, el nom de la font (per exemple: si la direcció de la font és a www.exemple.com, s'ha de posar 'exemple', ja que això ens servirà en el procés de recollida de les notícies). A partir d'aquí un petit codi s'ocupa de llegir el fitxer XML, d'extreure el nom i la descripció del bloc i emmagatzemar-ho tot a la base de dades.

En aquest mateix apartat també es pot trobar una llista amb totes les fonts d'informació prèviament introduïdes. Cal remarcar que el gestor ens permet fer varies operacions amb totes les fonts. Per una banda ens permet eliminar-les de la base de dades. També ens dona l'opció de veure i modificar els paràmetres que com s'ha explicat s'extreuen automàticament del fitxer XML. Per últim ens permet bloquejar-les, de manera que la font encara està a la base de dades però no s'utilitza en el procés de recollida de noves notícies. D'aquesta manera ens evitem esborrar-la de la base de dades, fent així que les seves dades es puguin utilitzar a l'hora de mostrar les notícies que s'han obtingut d'aquella font.

3.4.6.1.3. Menú: Ítems

Aquest grup d'opcions és el més important per l'aplicació, és en el que es troben totes les opcions relacionades amb les notícies.

En totes les opcions que apareixen es necessita introduir l'identificador únic de la notícia, tant per accedir a ella com a alguna de les seves parts. A continuació explicarem cada una de les opcions que tenim en aquest grup del menú.

A. Ítem

Al seleccionar aquesta opció del menú podem veure, eliminar i modificar una notícia. La intenció és no haver de modificar el contingut de les notícies, però es pot donar el cas que hi hagi algun error en la codificació d'algun caràcter. Per aquesta raó s'ha decidit poder modificar tant el títol com la descripció de cada ítem que tractem.

Una altra opció que ens permet és accedir a les altres informacions d'aquesta notícia per tal d'agilitzar el moviment per dintre de la notícia.

B. Tags

En aquest apartat es poden veure totes les paraules clau que defineixen una notícia. Aquesta informació és extreta de l'arxiu XML que ens aporta el canal web de cada font d'informació. De totes aquestes paraules tenim l'opció d'eliminar o modificar la paraula clau en qüestió.

El gestor ens permet també afegir més paraules a la llista, per tal de millorar el funcionament de l'aplicació.

C. Objectes HTML

Quan seleccionem aquesta opció el gestor ens permet veure, modificar o eliminar l'objecte *HTML* extret de la notícia. En la figura 3.11 podem veure com tenim un formulari per editar l'objecte per si hi ha aparegut algun error. Es mostra també l'objecte, i en el cas que

sigui un vídeo de *Youtube* es mostra la imatge de previsualització extreta mitjançant la seva pròpia API.

D. Relacionades

Es pot observar el grup d'identificadors de totes les notícies al qual pertany la notícia que estem buscant. Per tal de facilitar la feina, tots aquests identificadors són enllaços a la informació d'aquella notícia.

A més de llistar les relacions, ens permet modificar, eliminar o fins i tot afegir relacions al grup. En cas d'afegir l'identificador d'una notícia que té fills, aquests seran arrossegats amb ella i, per tant, també s'uniran al grup.

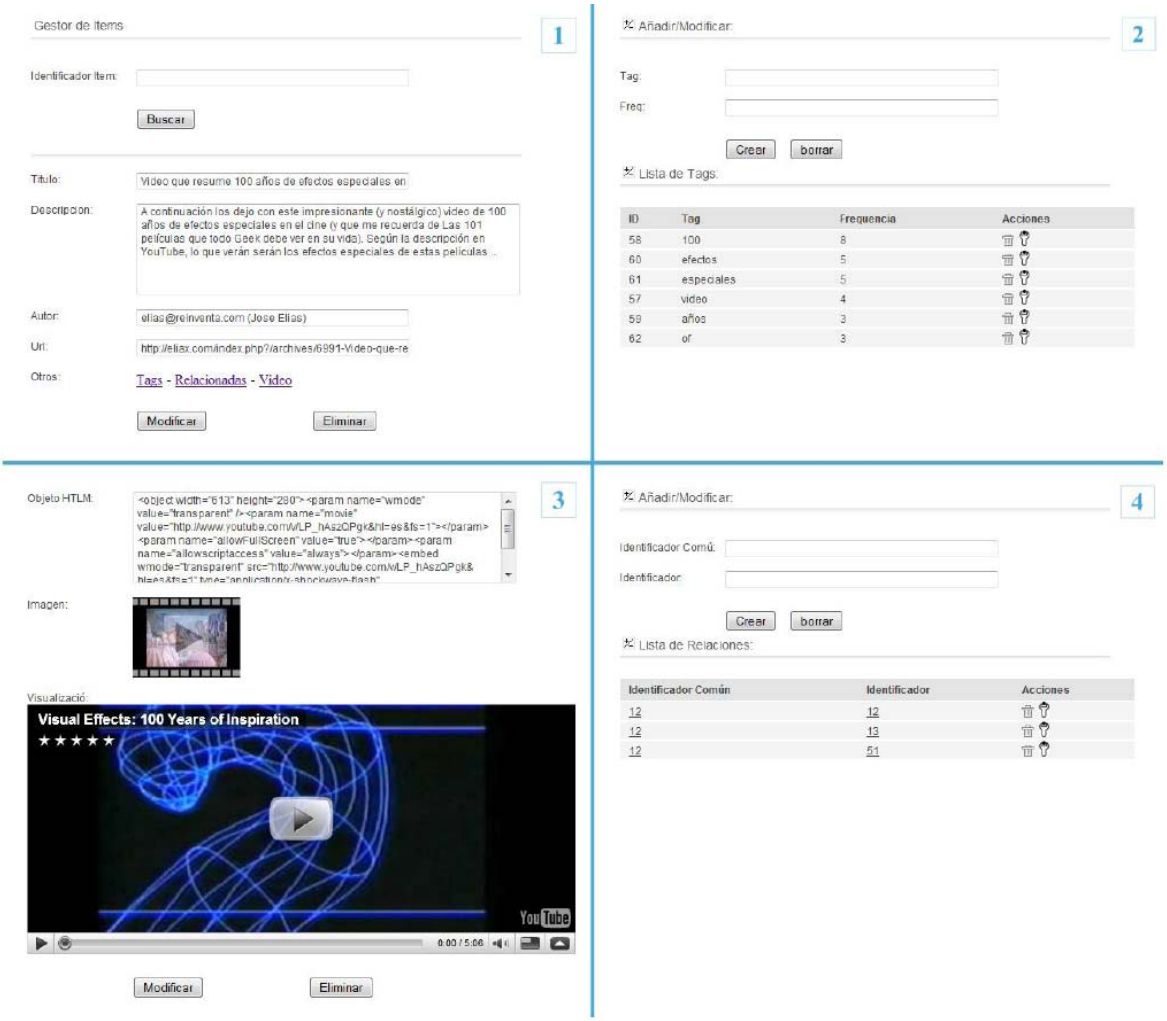

Figura 3.11. Opcions de l'apartat de Ítems

3.4.6.1.4. Menú: Diccionari

Aquest últim grup del menú és el que conté més opcions que ens permeten controlar algunes paraules o enllaços a l'hora de tractar i classificar les notícies. A continuació s'expliquen tots els seus apartats:

A. Paraules

Des d'aquest apartat es poden controlar totes les paraules que s'utilitzaran en els següents apartats. Per tal de tenir una base de dades organitzada i no tenir duplicats de paraules, s'ha decidit que totes les paraules abans de ser utilitzades s'han d'introduir en aquest apartat.

L'administrador podrà introduir totes les paraules que necessiti i a més a més, un cop afegides, les podrà eliminar o modificar.

B. Llistes blanques

En aquest apartat es mostren les llistes de paraules que ens permeten donar valors a paraules a l'hora d'analitzar i relacionar les notícies que les continguin.

Per tal d'introduir una paraula també s'ha d'indicar la temàtica a la que es vol aplicar, ja que s'ha considerat que el vocabulari de les diferents temàtiques pot variar. Per tant, poden existir tantes llistes com temàtiques tingui l'aplicació més la llista que anomenem 'Todas', la qual tindrà efecte per a totes les temàtiques de la nostra aplicació.

Com s'ha comentat al punt anterior, per introduir una paraula en aquestes llistes, aquesta ha d'haver estat introduïda prèviament a la llista de paraules.

C. Llistes negres

Aquestes llistes són un punt clau per fer la neteja dels textos, ja que són el conjunt de les paraules que s'eliminaran d'una notícia per així per les tasques d'una manera més òptima.

De la mateixa manera que a l'apartat anterior, les paraules que es vulguin introduir hauran d'existir en la llista de paraules i a més a cada inserció s'haurà d'indicar en quina temàtica es volen afegir.

Cal remarcar que existeix una llista negra per totes les temàtiques on es troben moltes de les paraules buides de significat en castellà.

D. Sinònims

Aquest apartat té un funcionament senzill. Ens permet d'introduir dues paraules en el formulari, prèviament afegides a la llista de paraules generals. D'aquesta manera aquelles dues paraules quedaran identificades com sinònimes.

Figura 3.12. Visió de l'opció 'Sinònims'

Per una paraula poden haver-hi diferents sinònims, com es pot observar en la figura 3.12**.** A més per fer una millor gestió d'aquesta opció permetem que cada conjunt de sinònims es pugui eliminar per complet, o també es pugui eliminar només una de les paraules sinònimes que s'hagin introduït seleccionant la icona oportuna.

E. Llista negra de URLs

Després de fer l'estudi de les parts de cada notícia, vam observar que era molt necessari crear aquest apartat, una llista que ens permeti bloquejar les direccions web introduïdes. La raó d'això va ser que molts dels blocs utilitzaven serveis de publicitat com per

exemple el de Google, i això fa que tots aquests hagin de posar un codi que conté la mateixa direcció web, fent que l'algorisme de relació tingui en compte aquesta direcció.

A la figura 3.13 es poden apreciar dues direccions que utilitzen els blocs per tenir publicitat a les seves notícies i que podríen fer funcionar erròniament el nostre algorisme de relació.

Figura 3.13. Llistes negres de URL

Aquesta secció ens permet eliminar, modificar o afegir en aquesta llista direccions web que siguin oportunes de bloquejar, per exemple si es dona el cas que apareguin altres serveis de publicitat.

3.4.6.2. Sistema de consulta de notícies

Aquesta és la part del projecte, en el qual es poden veure tots els resultats que es vagin produint. A més a més conté les funcionalitats demanades en els requeriments del projecte.

En les següents seccions s'explicaran les parts i funcionalitats que té l'aplicació.

3.4.6.2.1. Pàgina principal

En aquesta pàgina es mostraran les quinze notícies més actuals que l'aplicació hagi recollit i relacionat, per tal de mostrar la informació més actual.

Els criteris seguits per mostrar aquesta informació han sigut que es mostrin els grups de notícies relacionades indexades més recentment, ordenades pel número de relacions, és a dir el nombre de notícies d'aquell grup, i la seva data.

En aquesta mateixa pàgina, a la part dreta, podem trobar les diferents opcions que integra l'aplicació. En aquesta part de la pàgina principal ens permet escollir entre les diferents temàtiques i en quin idioma volem que estiguin escrites les notícies.

A la figura 3.14 es pot veure l'estructura que s'utilitza en la pàgina principal de l'aplicació, s'ha de tenir en compte, com ja s'ha dit en apartats anteriors, que no s'ha treballat en el disseny i per tant no és una aplicació atractiva. Primerament trobem cada notícia per separat on hi mostrem el títol de la notícia i l'autor. Seguidament trobem un petit fragment del cos de la notícia, del qual s'ha decidit publicar només un 25% o un màxim de 50 paraules, ja que no es pretenia replicar la informació dels blocs, sinó actuar de filtre relacionant tots aquells que parlen del mateix tema.

Figura 3.14. Plana principal de l'aplicació

Un cop mostrada tota la informació de la notícia principal del grup, trobem la informació referent a la resta de components del grup. Podem veure sota la notícia els títols de les altres notícies considerades com a iguals, juntament amb la informació del bloc que les ha publicat, ordenades per la seva data de publicació.

Per últim podem observar una finestra per compartir la notícia, on hi podem trobar la direcció d'aquest grup de notícies, on es mostrarà informació addicional. A més a més s'ha creat un

sistema on es pot compartir la notícia via correu electrònic o per dues grans xarxes socials: *facebook* i *twitter*.

Les dues xarxes ens permeten compartir enllaços, en cas de que l'usuari en formi part. Per exemple l'*API* de *facebook* ens permet utilitzar la següent comanda per poder publicar automàticament la notícia, juntament amb tota la informació en el perfil de l'usuari:

http://www.facebook.com/share.php?u=http://www.projecte.com/id_noticia

On *id_noticia* és la identificació de la notícia per situar-la en la nostra pròpia base de dades.

Twitter en canvi ens deixa compartir textos de fins a 140 caràcters, en comptes de direccions web. Per tant abans s'ha de crear un missatge a enviar. Es van provar diverses maneres, però la millor, vist el límit imposat per *twitter* ha sigut la de posar el titol i la direcció web de la notícia, com mostra el següent exemple.

Modernistas. Receta - http://tinyurl.com/6kd37n8

En aquest missatge es poden veure dues parts separades per un guió:

1. El títol de la notícia principal del grup
2. Una direcció web. Aquesta no és més que la direcció on trobem la notícia, però s'ha convertit en una direcció més curta, utilitzant l'*API* del servei *tinyurl* que ens permet l'escurçament de direccions web. D'aquesta manera reduïm la direcció i el missatge ens queda més curt i així l'usuari por aprofitar els caràcters restants per poder opinar sobre la notícia.

Per tant, la comanda que s'ha de fer en aquest cas és la següent:

http://twitter.com/home?status=Modernistas.%20Receta%20-%20http://tinyurl.com/6kd37n8

Capítol 4

Proves i experimentació

En aquest capítol es presenten els casos de prova per l'entorn desenvolupat en el projecte.

4.1. Sistema de relació

Per tal de fer les proves i comprovar l'eficàcia del sistema de classificació és necessari buscar un nou conjunt de dades. La manera d'aconseguir-lo és efectuant els mateixos passos que els del punt 3.4.5.2., seguits pel procediment de l'apartat 3.4.5.3. Un cop realitzats aquests passos obtenim un nou conjunt de dades amb un etiquetatge bastant fiable. A la taula 4.1 es poden observar les característiques d'aquest nou conjunt.

	Relacions positives	**Relacions negatives**
	40	15283
Total	15323	

Taula 4.1. Característiques del nou conjunt de dades

El següent pas és utilitzar la funció cost, creada per fer l'algorisme genètic, la qual retorna el tant per cent d'error que resulta d'aplicar un vector de pesos amb un conjunt de dades.

El vector de pesos utilitzat és *[9'61, 3'5, 4'01, 5'988, 9'032, 9'213, 7'998]*, amb el qual a la fase d'entrenament s'ha obtingut un error del *0'0008* i ara, amb el segon conjunt de dades (test) s'ha obtingut un error de *0'0011*.

A la taula 4.2 podem observar la matriu de confusió que ha generat l'algoritme. La combinació de pesos utilitzada ens dona uns resultats de falsos positius molt satisfactori, però ens proporciona un número lleugerament elevat de falsos negatius. El sistema resultant d'aquest mètode es podria considerar bastant conservador.

		Actuals	
		Ítems iguals	Ítems diferents
Prediccions	Ítems iguals	26	2
	Ítems diferents	14	15281
Total		40	15283
Error		0.0011	

Taula 4.2. Matriu de confusió

Una de les causes de que el sistema sigui més conservador és que ha estat entrenat amb un conjunt de dades etiquetat per una persona, i com que el criteri per decidir si dues notícies s'han de relacionar és molt subjectiu, s'ha cregut preferible optar per una decisió més conservadora. D'aquesta manera la classificació és més selectiva aconseguint així filar més prim i obtenir una millor separació de les diferents notícies.

A continuació es mostren alguns exemples que ens poden fer veure aquesta subjectivitat.

1. *Apple Mac cumple 27 años* (Isopixel)
2. *Mac, el sistema operativo de Apple, cumple 27 años* (elgrupoinformatico)
3. *Apple LED Cinema Display de 27 pulgadas* (tengounmac)

Com podem veure els dos primers titulars tindrien clara la seva unió, ja que els dos parlen de l'aniversari des de que va néixer oficialment el primer Mac. Però amb el tercer exemple es podría dubtar si posar-lo en el mateix grup o no, ja que per una banda conté moltes paraules en comú, però per l'altra banda es pot veure com és una notícia que parla sobre el llançament al mercat de les noves pantalles de 27" del sistema *LED cinema Display* d'*Apple*.

4.2. Sistema de classificació d'idioma

En aquest cas s'utilitza un conjunt de notícies, extretes d'una base de dades pública de diaris nacionals i internacionals en format *portable document format* (*pdf*), per fer les proves i comprovar l'eficàcia del sistema de classificació. Cal dir que cada una d'aquestes notícies té una mitja de 800 paraules.

En aquest cas tenim la avantatge que el conjunt de test ja està etiquetat prèviament i per tant podem realitzar les proves oportunes. En la següent taula mostro les característiques d'aquest conjunt de dades:

	Castellà	**Anglès**
	1278	1193
Total	2471	

Taula 4.3. Característiques del nou conjunt de dades

Un cop tenim un conjunt de dades ben etiquetades el que hem de fer per aconseguir obtenir els resultats que ens permetin comprovar el correcte funcionament del sistema es realitza el pas descrit en el punt 3.4.3.3., en el que hi apliquem l'algoritme desenvolupat basta en *Naive Bayes Multinomial*.

Per mostrar els resultats fem que l'algoritme generi una matriu de confusió, que mostrem en la taula 4.4. L'aplicació de la *Regla de Bayes* ens dóna una resultats molt satisfactoris, ho demostra el baix número de classificacions errònies que apareixen.

		Algoritme	
		Castellà	Anglès
Text	Castellà	1277	1
	Anglès	2	1191
Total		1279	1192

Taula 4.4. Matriu de confusió

Cal comentar que els dos idiomes utilitzats tenen un vocabulari molt diferent i per tant permet que obtinguem uns resultats tan satisfactoris. Un cop obtinguts aquests resultats el que hem fet és comprovar el motiu de que s'hagin classificat malament tres textos. Al analitzar els tres textos ens hem donat compte que el text que classifica com a anglès tot i estar etiquetat en castellà està format per molt acrònims en anglès que fan classificar incorrectament el text. Pel que fa als dos textos classificats en castellà tot i estar etiquetats en anglès el que ens hem donat compte és que estan etiquetats erròniament ja que pertanyen a una font d'informació anglesa que ha etiquetat la notícia en anglès tot i estar el text en castellà.

Per tant havent escollit aquests dos idiomes per classificar textos podem dir que té un 100% d'encert en les proves que hem realitzat.

4.3. Sistema de classificació per temàtica

L'anàlisi dels resultats d'aquest sistema es separaren depenent si les fonts són en anglès o si són en castellà ja que depenent d'aquesta classificació utilitzarem un algoritme o un altre.

En el moment de fer l'entrenament havíem utilitzat un total de 500 notícies per cada una de les temàtiques i per cada un dels idiomes. En aquest moment el conjunt de dades és més gran ja que hem emmagatzemat més notícies degut a la diferència de temps i degut també a que hem augmentat el número de fonts d'informació utilitzades en la recol·lecció de notícies.

4.3.1. Conjunt de dades en castellà

Per tal de fer les proves i comprovar l'eficàcia del sistema de classificació tenim un conjunt de dades que mostro en la taula 4.3.

	Motor	**Cuina**	**Fotografia**
Castellà	1127	894	1033
Total	3054		

Taula 4.5. Característiques del nou conjunt de dades

Un cop tenim un conjunt de dades ben etiquetat el següent pas a realitzar és aplicar l'algoritme explicat en el punt 3.4.4.3.3. Aquest algoritme ens compara el resultat que obté amb l'etiquetatge original del conjunt de dades. Per poder observar el funcionament de l'algoritme s'han separat els resultats segons l'etiquetatge previ del conjunt de dades, separant així els resultats per temàtiques. En les taules 4.4, 4.5 i 4.6 podem observar com augmenta l'encert segons el grup de dades augmenta.

		Algoritme SVM
Número d'elements etiquetats	**1127**	89.17%
	1000	88.43%
	500	86.23%
	200	82.11%
	100	73.54%
	50	62.06%

Taula 4.6. Resultats d'aplicar SVM al conjunt de *Motor*

		Algoritme SVM
Número d'elements etiquetats	**894**	86.33%
	700	85.49%
	400	83.78%
	200	78.71%
	100	69.12%
	50	59.34%

Taula 4.7. Resultats d'aplicar SVM al conjunt de *Cuina*

		Algoritme SVM
Número d'elements etiquetats	**1033**	89.32%
	1000	89.13%
	500	87.74%
	200	83.29%
	100	74.96%
	50	63.03%

Taula 4.8. Resultats d'aplicar SVM al conjunt de *Fotografia*

Com podem observar els millors resultats ens el dóna l'algoritme quan l'apliquem al conjunt de dades de la classe *Fotografia*, seguit per la classe *Motor*, això és degut a que el vocabulari utilitzat és més tècnic en el cas d'aquestes dues classes i per tant pot classificar més eficaçment.

4.3.2. Conjunt de dades en anglès

En aquest cas mostrem en la taula 4.9 les característiques de les dades utilitzades per comprovar el correcte funcionament del sistema de classificació.

	Motor	**Cuina**	**Fotografia**
Anglès	1789	1237	1652
Total	4678		

Taula 4.9. Característiques del nou conjunt de dades

Com podem observar el número de notícies recopilades en aquest idioma és superior a l'anterior, això és degut al gran nombre de fonts d'informació que tenim a l'abast en aquest idioma.

Com en el cas anterior, en el que tenim el conjunt de dades en castellà, un cop tenim el conjunt de dades ben etiquetat el següent pas a realitzar és aplicar l'algoritme explicat en el punt 3.4.4.3.3. Aquest algoritme ens compara el resultat que obté amb l'etiquetatge original del conjunt de dades. Per poder observar el funcionament de l'algoritme s'han separat els resultats segons l'etiquetatge previ del conjunt de dades, separant així els resultats per temàtiques. En les taules 4.10, 4.11 i 4.12 podem observar com augmenta l'encert segons el grup de dades augmenta.

		Algoritme SVM
Número d'elements etiquetats	**1789**	91.53%
	1500	90.49%
	1000	89.74%
	500	86.42%
	200	82.07%
	100	72.98%
	50	61.84%

Taula 4.10. Resultats d'aplicar SVM al conjunt de *Motor*

		Algoritme SVM
Número d'elements etiquetats	**1237**	89.94%
	1000	88.03%
	500	84.89%
	200	79.37%
	100	71.59%
	50	60.21%

Taula 4.11. Resultats d'aplicar SVM al conjunt de *Cuina*

		Algoritme SVM
Número d'elements etiquetats	**1652**	91.72%
	1500	91.38%
	1000	90.89%
	500	88.67%
	200	84.71%
	100	75.59%
	50	63.65%

Taula 4.12. Resultats d'aplicar SVM al conjunt de *Fotografia*

Al observar aquestes dades ens donem compte que l'encert respecte l'altre idioma és lleugerament millor. Els resultats que obtenim no són del tot acceptables, però és degut a la quantitat de dades que podem utilitzar per millorar l'algoritme. Al augmentar la nostre base de dades podrem millorar els resultats i oferir uns resultats molt més satisfactoris.

Capítol 5

Conclusions i línies obertes

Fent una revisió dels objectius inicials del projecte i vist el rendiment obtingut pels algoritmes i l'aplicació web implementada, en aquest capítol es procedirà a extreure unes conclusions generals. Per altra banda, aquestes es completaran amb l'explicació del possible treball futur que es pot fer per millorar el funcionament de l'aplicació.

5.1. Conclusions

En aquest treball s'ha realitzat una aplicació que incorpora un conjunt d'algoritmes que permeten la classificació de notícies provinents de la xarxa en idioma i temàtica i un cop fet això es relacionen entre elles per formar grups de notícies que parlen sobre el mateix tema.

Davant de la problemàtica de la classificació per idioma i per temàtica, la qual s'ha considerat una tasca semisupervisada i multiclasse, s'ha vist la necessitat de proposar tècniques que s'adaptessin a aquest entorn, s'ha vist la necessitat, en el cas de la classificació per temàtica, de proposar tècniques que s'adaptin a aquest entorn, ja que *SVM* únicament soluciona problemes supervisats i binaris per naturalesa. Pel cas de la classificació per temàtica s'ha utilitzat la *Regla de Bayes* per desenvolupar un *Naive Bayes Multinomial* que separant en bigrames les dades introduïdes ens han aportat uns resultats molt satisfactoris.

Cal comentar que des del primer dia s'ha tingut molt en compte el futur del projecte, per això es va decidir crear una eina modulable i fàcil de modificar. Per aquesta raó la creació de diverses classes que tracten tot els tipus de dades de l'aplicació ha estat clau per aconseguir aquest objectiu i així poder afrontar més fàcilment les millores futures que es puguin aplicar.

L'aplicació tot i estar en local ha estat en funcionament recol·lectant notícies contínuament i s'ha pogut comprovar, en funcionament, les bones relacions i la correcta classificació de notícies i per això podem dir que s'ha desenvolupat un sistema que ens aporta un servei eficient.

5.2. Treball futur

Durant el desenvolupament del projecte han anat apareixent detalls i idees que hagués agradat implementar. Si no ha estat així es per no voler sortir de la línia de planificació inicialment marcada. Les possibles millores a aplicar al projecte s'expliquen a continuació.

El primer punt que s'ha de treballar és el disseny de l'aplicació per poder convertir-la en una aplicació atractiva i per tant pública. Això ens ajudaria a analitzar l'evolució de l'aplicació i poder rebre consells i queixes dels usuaris per seguir treballant en una millor aplicació.

Una de les millores que s'havia pensat és la opció de la compartició de notícies i permetre realitzar comentaris en cada una de les notícies, fent així una aplicació col·laborativa i participativa. Un altre punt a valorar seria el desenvolupament d'un sistema d'extracció de paraules clau de les notícies per buscar informació addicional de la noticia en buscadors o en xarxes socials. Una solució a desenvolupar seria donar valors sintàctics a les paraules fent així que les diferents paraules tinguin diferents valors, per exemple les paraules en negreta, cursiva o paraules que defineixen enllaços podrien tenir més pes a l'hora de ser analitzades. A més a més aquesta solució també ens seria útil en el moment de relacionar notícies.

La creació de comptes premium a disposició dels usuaris seria un gran avenç en el projecte. D'aquesta manera es permetria que aquests poguessin indicar les fonts d'informació, idioma i temàtica que els interessen i el sistema de relació i classificació desenvolupat treballaria amb aquestes, fent així que cada usuari tingues uns resultats personalitzats segons les fonts triades. Un altre avantatge d'aquesta millora esdevindria degut a que els usuaris són una font molt gran per la generació d'estadístiques.

Una altra petita millora seria fer utilitzar unes URIs més intuïtives, és a dir, en comptes d'utilitzar l'identificador, fer servir el títol de la notícia, de manera que les direccions serien més intuïtives i l'enllaç ja aportaria informació sobre el que tracta la notícia.

I per últim, una altra de les millores importants seria la d'ampliar les opcions de temàtica i la de d'idiomes amb les que estan escrites les notícies, fent d'aquesta una aplicació molt més competitiva en l'amplia xarxa d'Internet.

Capítol 6

Bibliografia i referències

- Wikipedia.org. Support Vector Machine. [Actualitzada el 30 de Juny del 2010] Disponible a: http://en.wikipedia.org/wiki/Support_vector_machine
- Nello Cristianini and John Shawe-Taylor. An Introduction to Support Vector Machines and other kernel-based learning methods. Cambridge University Press, 2000.
- Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin. A Practical Guide to Support Vector Classification. Taipei: Department of Computer Science: National Taiwan University; 2003- [Actualitzat el 15 d'Abril del 2010]. Disponible a: http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf
- Support-vector-machines.org. SVM – Support Vector Machines: Software. 2007. Disponible a: http://www.support-vector-machines.org/SVM_soft.html
- Wikipedia.org. Bayes' theorem. [Actualitzada el 29 de Gener del 2011]. Disponible a: http://en.wikipedia.org/wiki/Bayes'_theorem
- Wikipedia.org. Bayesian probability. [Actualitzada el 10 de Gener del 2011]. Disponible a: http://en.wikipedia.org/wiki/Bayesian_probability
- Wikipedia.org. Naive Bayes classifier. [Actualitzada el 21 de Gener del 2011]. Disponible a: http://en.wikipedia.org/wiki/Naive_Bayes_classifier
- Sebastian, Fabrizio. Machine Learning in Automated Text Categorization. Italia: Consiglio Nazionale delle Ricerche.
- Segaran, Toby. Programming Collective Intelligence. Estats Units d'Amèrica: O'Reilly Media; 2007
- W. Moore, Andrew. Naive Bayes Classifier. School of Computer Science: Carnegie Mellon University. 2004. Disponible a: http://www.autonlab.org/tutorials/naive02.pdf
- SimplePie.org. Simple Pie. Disponible a: http://simplepie.org
- PHP.net. PHP: Hypertext Preprocessor. Disponible a: http://php.net/
- Cabezas Granado, Luis Miguel. Manual imprescindible de PHP 5. Madrid: Anaya Multimedia; 2004.
- Wikipedia.org. Web Feed. [Actualitzada el 9 de Gener del 2011]. Disponible a: http://en.wikipedia.org/wiki/Web_feed

- Wikipedia.org. RSS. [Actualitzada el 16 de Gener del 2011]. Disponible a: http://en.wikipedia.org/wiki/RSS
- Wikipedia.org. Atom. [Actualitzada el 12 de Gener del 2011]. Disponible a: http://en.wikipedia.org/wiki/Atom_(standard)
- Intertwingly.net. Rss20AndAtom10Compared. Disponible a: http://www.intertwingly.net/wiki/pie/Rss20AndAtom10Compared
- Intertwingly.net. RSS Quick summary. Disponible a: http://www.intertwingly.net/slides/2003/rssQuickSummary.html
- Tbray.org. Atomic RSS. [Actualitzada el 28 de Juliol del 2005]. Disponible a: http://www.tbray.org/ongoing/When/200x/2005/07/27/Atomic-RSS
- Wikipedia.org. XML – Extensible Markup Language. [Actualitzada el 27 de Gener del 2011]. Disponible a: http://en.wikipedia.org/wiki/XML
- W3C.es. Guía Breve de Tecnologías XML. [Actualitzada el 9 de Gener del 2008]. Disponible a: http://www.w3c.es/divulgacion/guiasbreves/tecnologiasxml
- MySQL.com. Full-Text Search Functions. Disponible a: http://dev.mysql.com/doc/refman/5.0/en/fulltext-search.html
- Wikipedia.org. ISO/IEC 8859.1. [Actualitzada el 26 de Gener del 2011]. Disponible a: http://en.wikipedia.org/wiki/ISO/IEC_8859-1
- Wikipedia-org. UTF-8. [Actualitzada el 28 de Gener del 2011]. Disponible a: http://en.wikipedia.org/wiki/UTF-8
- Padró, Lluís. Llista de paraules buides de contingut en castellà. UPC. Disponible a: http://www.lsi.upc.es/~padro/freqs/empty.sp.2.gz
- Sourceforge.net. OSU-SVM. Disponible a: http://svm.sourceforge.net/docs/3.00/api/
- Martí i Gòdia, Enric. Bases de Dades I. Setembre 2006.
- Vitrià, Jordi. Intel·ligència Artificial II – Apunts de l'assignatura. 2001.

Annexos

A.1. Fonts d'informació

Fonts d'informació sobre la temàtica Cuina

http://www.directoalpaladar.com	http://www.7canibales.com
http://www.foodmall.org	http://www.gastronomiaycia.com
http://www.blogexquisit.com	http://lefabuleuxdestinduchocolat.blogspot.com
http://www.aliterdulcia.com	http://asopaipas.blogspot.com
http://nikesamo.blogspot.com	http://cogollosdeagua.blogspot.com
http://www.javirecetas.com/	http://lasopagansa.blogspot.com
http://recetasytragos.com	http://www.lacocinadelechuza.com
http://www.muchogusto.net	http://www.lasrecetasdesara.com
http://bocadosdulcesysalados.blogspot.com	http://larsvontrier.blogspot.com
http://www.marialunarillos.com	http://www.elzurrondelospostres.com
http://www.recetasdemama.es	http://www.lacocinadeauro.com
http://webosfritos.es	http://chocolatepimienta.blogspot.com
http://www.elrincondebea.com	http://www.midulcetentacion.es
http://elrincondelamariposa.blogspot.com	http://panyvarios.blogspot.com

http://www.foodandcook.net	http://martzela223.wordpress.com
http://simplyrecipes.com/	http://www.spittoon.biz
http://www.tarteletteblog.com/	http://www.acookblog.com/
http://thepassionatecook.typepad.com/	http://thecookingblog.blogspot.com/
http://www.tvcocina.com/	http://www.cocinaparahombres.com/
http://www.guisando.org/	http://laollasuiza.blogspot.com/
http://bocadorada.com/	http://zuccheriera.blogspot.com/
http://biscotti.blogspot.com	http://panepizza.blogspot.com/
http://acibecheria.blogspot.com/	http://www.recetas-cocina.com.es/
http://lodecarlosvalencia.blogspot.com/	http://www.falsariuschef.com/
http://ondakin.com/	http://hechoencocina.blogspot.com/
http://elcocinerofiel.com/	http://cocinarparalosamigos.blogspot.com/
http://www.condelantal.com/	http://pecadosdelmonaguillo.blogspot.com/
http://garbancita.blogspot.com/	http://nachovazquez.blogspot.com/
http://gourmetymerlin.blogspot.com/	http://www.daviddejorge.com/
http://www.martinberasateguiblog.com/	http://manuelallue.blogspot.com/
http://www.elgranchef.com/	http://saborgourmet.com/

http://www.colineta.com/	http://observaciongastronomica.blogspot.com/
http://www.eladerezo.com/	http://www.motherearthnews.com/
http://cannelle-vanille.blogspot.com/	http://rasamalaysia.com/
http://trissalicious.com/	http://almostbourdain.blogspot.com/
http://www.davidlebovitz.com/	http://www.latartinegourmande.com/
http://www.thekitchn.com/	http://dinersjournal.blogs.nytimes.com/
http://www.acupcakeortwo.com/	http://ooh-look.blogspot.com/
http://www.figandcherry.com/	http://simonfoodfavourites.blogspot.com/
http://grabyourfork.blogspot.com/	http://www.jenius.com.au/
http://citrusandcandy.com/	http://www.atablefortwo.com.au/
http://www.foodmall.org/	

Fonts d'informació sobre la temàtica Motor

http://8000vueltas.com	http://www.motorpasion.com
http://www.diablomotor.com	http://www.diariomotor.com
http://trend-tech.blogspot.com/	http://www.cardesign.tv/
http://thenextgear.com/	http://artadytha.blogspot.com/
http://www.ameinfo.com/	http://europeanmotornews.com/
http://www.autocar.co.uk/	http://www.autoblog.com/
http://www.jalopnik.com/	http://www.autoexpress.co.uk/
http://www.autozeitung.de/	http://www.thetruthaboutcars.com/
http://www.engadget.com/	http://carscoop.blogspot.com/
http://www.autoweek.com/	http://carscoop.blogspot.com/
http://www.leftlanenews.com/	http://www.caranddriver.com/
http://rss.edmunds.com/	http://www.wheels.ca/
http://www.autonews.com/	http://www.just-auto.com/
http://www.just-auto.com/	http://blogs.edmunds.com/
http://blogs.edmunds.com/	http://blogs.edmunds.com/karl/
http://blogs.edmunds.com/greencaradvisor/	http://blogs.consumerreports.org/cars/
http://rss.edmunds.com/IL/	http://fastlane.gmblogs.com/
http://www.autospies.com/	http://germancarscene.com/
http://www.eurocarblog.com/	http://www.autospies.com/

http://www.leftlanenews.com/	http://www.egmcartech.com/
http://blogs.edmunds.com/straightline/	http://www.topspeed.com/
http://www.topgear.com/	http://www.whatcar.com/
http://rss.autotrader.co.uk/	http://www.parkers.co.uk/
http://www.telegraph.co.uk/	http://www.pistonheads.com/
http://www.autosport.com/	http://www.planet-f1.com/
http://www.evo.co.uk/	http://www.f1fanatic.co.uk/
http://www.pitpass.com/	http://www.grandprix.com/
http://www.f1technical.net/	http://www.f1racing.net/
http://www.formula1.com/	http://news.bbc.co.uk/sport2/hi/motorsport/
http://machinespider.com/	http://www.newstechnologyautomotive.com/
www.formula1hoy.com/	http://www.notodocoches.com/
http://www.infocoches.com/	http://www.motorspain.com/
http://www.tecnocoches.com/	http://todosobref1.blogspot.com/
http://www.hoymotor.com/	http://motor.terra.es/
http://www.highmotor.com/	http://www.actualidadmotor.com/
http://es.motorfull.com/	http://es.autoblog.com/
http://www.racingpasion.com/	http://www.circulaseguro.com/
http://www.actualidadmotor.com/	http://www.cochesafondo.com/
http://www.revistamotoviva.com/	http://scooters-y-ciclomotores.dailymotos.com/

http://www.motociclismo.es/	http://www.lamoto.es/
http://www.visordown.com/	http://www.motorcyclenews.com/
http://www.motorcycle.com/	http://motorcycles.about.com/
http://thekneeslider.com/	http://www.motorcycle-usa.com/
http://roadracingworld.com/	http://motorcyclebloggers.com/
http://carlaking.typepad.com/	http://motomatters.com/
http://londonbikers.com/	http://blogs.lavozdegalicia.es/pumarola/

Fonts d'informació sobre la temàtica Fotografia

http://haciendofotos.com/	http://www.ojodigital.com/
http://foto.microsiervos.com/	http://www.parasaber.com/tecnologia/fotografia-digital/
http://haciendofotos.com/	http://www.jggweb.com/
http://www.dsmcomunicacion.net/	http://www.fotomaf.com/blog/
http://memoflores.com/	http://www.enfocando.es/
http://fogonazos.blogspot.com/	http://www.caborian.com/
http://www.canonistas.com/	http://www.elclubdigital.com/
http://escuelafotosevilla.blogspot.com/	http://javiergarciarosell.wordpress.com/
http://vampyressa.blogspot.com/	http://solofotography.blogspot.com/
http://www.photoshop-designs.net/	http://www.cofregrafico.com/
http://www.compartetusrecuerdos.com/	http://www.comolahice.com/
http://procesocruzado.com/	http://naturpixel.com/
http://www.ignacioizquierdo.com/	http://www.fotonatura.org/
http://from10to300mm.pixyblog.com/	http://www.chromasia.com/
http://fotografsnatura.blogspot.com/	http://fotodenatura.blogspot.com/
http://xaviheredia.aminus3.com/	http://mute.rigent.com/
http://10mmgalore.com/	http://www.markpower.me.uk/
http://www.justingaynor.com/	http://www.krisvdv.net/pixelpost/
http://www.photoschau.de/	http://moodaholic.com/

http://www.durhamtownship.com/	http://www.dpreview.com/
http://www.mylalaland.com/hello/	http://www.filemagazine.com/
http://martinandreasen.dk/	http://www.pixeldreamer.de/
http://www.sirius2photo.com/	http://framedandshot.com/
http://sobrefotos.com/	http://www.somedaysomewhere.net/
http://www.rock-climb.de/	http://www.pearweed.net/
http://www.ottokphotography.com/	http://bluechameleon.aminus3.com/
http://beanow.alkos.info/	http://www.markushartel.com/
http://strobist.blogspot.com/	http://www.joemcnally.com/blog/
http://www.scottkelby.com/blog/	http://feeds.feedburner.com/ChaseJarvis
http://lightroomkillertips.com/	http://www.lbecker.com/blog/
http://www.zarias.com/	http://www.digital-photography-school.com/
http://theonlinephotographer.typepad.com/	http://feeds.feedburner.com/In-public
http://elainev.com/	http://www.strawberryfields.pl/
http://damianchrobak.blogspot.com/	http://photoblog.jbuhler.com/
http://www.perspective-images.com/	http://yzblog.hu/
http://friskypics.com/	http://xrp-photoblog.blogspot.com/
http://wvs.topleftpixel.com/	http://www.deceptivemedia.co.uk/
http://fotoaprendiz.com/	http://www.nuevafotografia.com/
http://www.xatakafoto.com/	http://www.fotografia.com/

Signat: Xavier Rabadán Rius
Bellaterra, 31 de Gener del 2011

Resum

El projecte es centra en el desenvolupament d'un recol·lector de notícies publicades a una llarga llista de blocs ampliada contínuament pel desenvolupador i pels usuaris, afegint els seus blocs preferits. L'aplicació desenvolupada realitza una recol·lecció continua de notícies consultant les possibles novetats que apareguin en cada un dels blocs inscrits a l'aplicació. Se'ls hi aplica un classificador per idioma i per temàtica i es relaciona amb les altres notícies existents si aquestes parlen sobre el mateix tema. En l'aplicació desenvolupada hi ha la possibilitat d'escollir entre les temàtiques ofertes i en l'idioma que ha estat publicada la notícia. Pel desenvolupament del projecte s'ha desitjat que la plataforma sigui el més compatible possible amb la tecnologia actual fent servir diversos llenguatges de programació que han permès desenvolupar cada un dels algorismes necessaris pel desenvolupament global de l'aplicació; en ordre d'ús he fet servir Php, Matlab, Html, MySql, CSS3, Javascript i XML. S'ha de destacar que el projecte aporta una comoditat per tots aquells lectors de blocs que es troben tantes vegades amb notícies ja llegides en els diferents blocs que consulten.

Resumen

El proyecto se centra en el desarrollo de un recolector de noticias publicadas en una larga lista de bloques ampliada continuamente por el desarrollador y los usuarios, añadiendo sus blogs favoritos. La aplicación desarrollada realiza una recolección continua de noticias consultando las posibles novedades que aparezcan en cada uno de los blogs inscritos en la aplicación. Se les aplica un clasificador de idioma y por temática y se relaciona con las otras noticias existentes si estas hablan sobre el mismo tema. En la aplicación desarrollada está la posibilidad de escoger entre las temáticas ofrecidas y en el idioma que ha sido publicada la noticia. Para el desarrollo del proyecto se ha deseado que la plataforma sea lo más compatible posible con la tecnología actual utilizando varios lenguajes de programación que han permitido desarrollar cada uno de los algoritmos necesarios para el desarrollo global de la aplicación, en orden de uso he usado Php, Matlab, Html, MySql, CSS3, Javascript y XML. Cabe destacar que el proyecto aporta una comodidad para todos aquellos lectores de blogs que se encuentran tantas veces con noticias ya leídas en los diferentes blogs que consultan.

Abstract

The project focuses on developing a collection of news published in a long list of blogs continuously extended by the developer and users, adding their favorite blogs. The developed application performs a continuous collection of news referring to the possible developments that appear in each of the blogs listed in the application. This application applies a language and theme classifier and relates to other existing news if treat about the same topic. In the developed application you can choose among the topics offered and the language that has been published the news. Project development has been desired that the platform is as compatible as possible with current technology using various programming languages that have allowed for each of the algorithms necessary for the overall development of the application; in order of use I used Php, Matlab, Html, MySql, CSS3, JavaScript and XML. Notably, the project provides comfortfor those blog readers who find lots of news already read.