Gobierno del Principado de Asturias
Consejería de Economía y Empleo

Unión Europea
Fondo Europeo
de Desarrollo Regional

RESULTADOS AIDOS

Análisis de Información Desestructurada, Opiniones y Sentimientos

En Gijón , a 31 de Diciembre de 2017

1. MEMORIA TÉCNICA

Introducción

El proyecto AIDOS (Análisis de Información Desestructurada, Opiniones y Sentimientos) ha sido ejecutado por Fundación CTIC – Centro Tecnológico entre el 1 de enero de 2016 y el 31 de diciembre de 2017. **Esta memoria recoge la descripción de los trabajos realizados a lo largo de todo el proyecto,** integrando los trabajos ya recogidos en el informe de seguimiento de la anualidad 2016 con los desarrollados en la anualidad 2017.

El objetivo del proyecto AIDOS consiste en el diseño e implementación de un sistema con capacidad para procesar mediante técnicas y algoritmos de *Data Science* el lenguaje natural, lo que permitirá la recogida, integración, análisis y clasificación de información desestructurada, así como establecer perfiles de opinión y sentimientos respecto a una temática en concreto. Así, este proyecto se divide en dos bloques principales: un "analizador de texto" encargado de realizar el análisis semántico del texto libre, eliminando el ruido (mensajes no relevantes para una prospectiva de mercado) y detectando las menciones a entidades comerciales (empresas, productos y marcas); y un "identificador de perfiles" encargado de establecer perfiles de opinión (o *clusters*) de comunidades y usuarios, construyendo una fuente de información que los expertos de marketing puedan utilizar para estudiar la percepción, valoración y reacción social de la temática analizada.

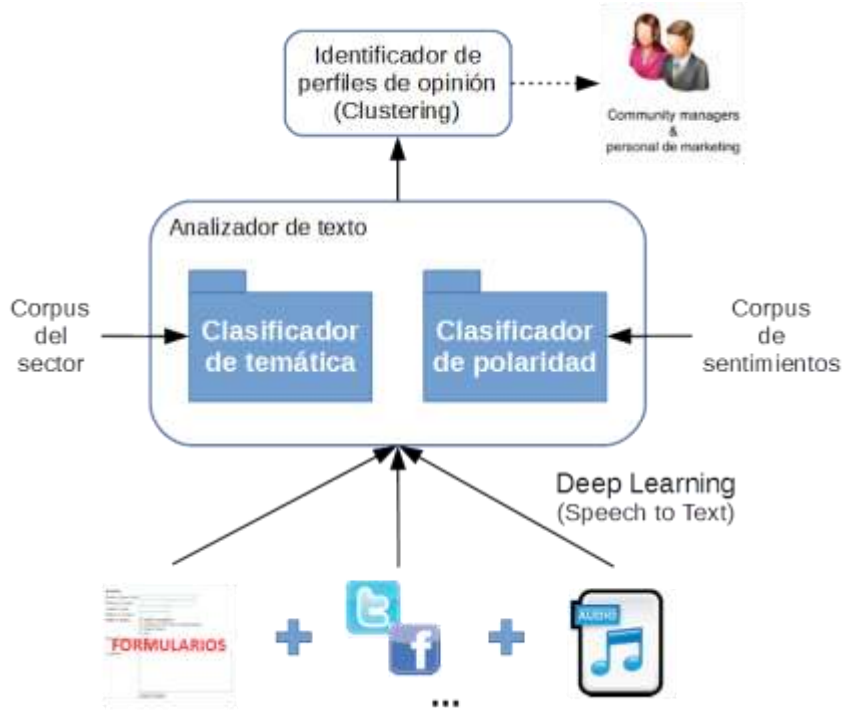

Figura 1: Esquema general del sistema analizador de texto

La ejecución del proyecto AIDOS se ha dividido en cinco hitos claramente diferenciados:

- Hito 1: Análisis y diseño del sistema (realizado en la anualidad 2016).
- Hito 2: Análisis de texto libre (realizado en la anualidad 2016).
- Hito 3: Identificador de perfiles de opinión (iniciado en la anualidad 2016 y finalizado en la 2017).
- Hito 4: *Deep Learning (*iniciado en la anualidad 2016 y finalizado en la 2017).
- Hito 5: Integración y validación del sistema en el sector de la alimentación (realizado en la anualidad 2017).

A continuación, se describen los trabajos realizados en cada uno de los hitos mencionados anteriormente y los resultados conseguidos.

ANUALIDAD 2016

Hito 1: Análisis y diseño del sistema

El objetivo principal de este hito fue tanto la especificación de requisitos del sistema como el diseño técnico del mismo, desglosando la actividad en dos tareas principales:

- T1.1: Análisis de requisitos
- T1.2: Diseño y arquitectura de la solución propuesta

T1.1: ANÁLISIS DE REQUISITOS

A lo largo de la primera tarea se han investigado y detallado los requisitos mínimos que debe reunir la herramienta desarrollada para que pueda ser implantada en aquellas aplicaciones en las que el análisis de la información desestructurada puede aportar importantes ventajas y beneficios a la empresa.

Se han identificado los principales requisitos del sistema, tanto funcionales como no funcionales, desde el punto de vista del almacenamiento y procesamiento de datos y la prioridad en cuanto a su aplicación (Alta/Obligatorio, Media/Recomendable, Baja/Opcional).

A continuación, se presentan los requisitos funcionales (RF) y no funcionales (RNF) del sistema que se han identificado:

Adquisición e integración de datos		
RF-1	El sistema debe de ser capaz de adquirir e integrar información desestructurada proveniente de diversas fuentes de información, tales como encuestas, formularios, grabaciones, etc.	Alta

RF-2	El sistema debe de ser capaz de adquirir información de la red social *Twitter* analizando las posibilidades de interacción que se puede obtener de esta plataforma.	Alta
RF-3	El sistema desarrollado debe ser capaz de analizar contenido en formato audio y transformarlo a formato de texto (*Speech to text*), garantizando unos niveles mínimos de calidad en la transcripción, es decir, perdiendo o confundiendo la menor cantidad de información posible.	Alta
RF-4	El sistema desarrollado debe ser capaz de exportar la transcripción de audio obtenida en formato de texto, para su posterior análisis con el resto de técnicas desarrolladas.	Alta
RF-5	El sistema desarrollado deberá homogeneizar e integrar en la arquitectura la información recogida para su posterior análisis.	Alta
RF-6	El sistema de análisis debe de ser independiente de la naturaleza origen de los datos.	Alta
Identificación de temática		
RF-7	El sistema desarrollado debe ser capaz de reconocer la temática abordada o entidad mencionada en el texto analizado.	Alta
Identificación de la polaridad		
RF-8	El sistema desarrollado debe ser capaz de extraer la polaridad de opinión referente a la temática abordada o entidad mencionada, permitiendo extraer el grado de satisfacción o posicionamiento del usuario respecto a una determinada marca, producto o empresa.	Alta
***Clustering* de usuarios**		

RF-9	El sistema desarrollado debe ser capaz de extraer integrar y analizar toda la información recopilada para extraer la percepción social que existe respecto a una entidad determinada, identificando automáticamente diferentes perfiles de opinión o *clusters* de usuarios.	Alta
Presentación y almacenamiento de la información		
RF-10	El sistema desarrollado debe ser capaz de presentar los resultados obtenidos a través de un informe interpretable, que utilice gráficos y tablas que permitan la comprensión del análisis por parte de cualquier usuario no experto en la materia.	Alta
RF-11	El sistema desarrollado debe permitir recuperar la información almacenada que ha permitido al sistema extraer las conclusiones presentadas en el informe final (temática, polaridad, *clustering*, etc.)	Alta
RF-12	La recuperación de información se podrá realizar bajo diferentes niveles de granularidad (fuentes de origen, fecha de adquisición, etc.)	Alta

Extrapolabilidad		
RNF-1	El sistema será extrapolable a cualquier sector comercial para el que el análisis de información desestructurada suponga una importante ventaja competitiva.	Alta
Interoperabilidad		
RNF-2	El sistema desarrollado deberá asegurar un alto grado de interoperabilidad utilizando estándares cuando sea posible.	Alta
	El sistema desarrollado deberá ser escalable vertical y horizontalmente, permitiendo ampliar su productividad.	Alta

RNF-3		
Interpretabilidad		
RNF-4	El sistema desarrollado debe permitir, en la medida de lo posible, que los usuarios finales interpreten la información recibida, de manera que quede justificada la elección de la polaridad y temática de los textos.	Alta
Rendimiento		
RNF-5	Los tiempos de ejecución de los algoritmos (detección de temática, polaridad, *Speech to text*, *clustering*) han de ser aceptables, compitiendo con el resto de algoritmos presentes en la literatura.	Alta

Tabla 1: Requisitos que debe cumplir el sistema desarrollado

T1.2: Diseño y arquitectura de la solución propuesta

Una vez analizados los requisitos generales de la aplicación a desarrollar se ha elaborado el diseño técnico de la solución. Para poder cumplir con los requisitos de la aplicación se ha decidido usar un diseño basado en los siguientes módulos:

1. **Módulo de adquisición e integración de información desestructurada:** este módulo es necesario para poder adquirir e integrar información desestructurada proveniente de diferentes fuentes de información

2. **Módulo de "*Speech to Text*":** este módulo es el encargado de recopilar la información que se presente en formato de audio y transformarla a texto, garantizando unos niveles mínimos de calidad en la transcripción.

   Para ello se están estudiando las diferentes herramientas existentes en el mercado. En particular, se está realizando un estudio detallado de las APIs más conocidas, como son *Google Speech Recognition*, *Microsoft Bing Voice Recognition* o *IBM Speech to Text*. Además de las anteriores, se está explorando la opción de desarrollar un módulo propio de "*Speech to Text*". Este módulo integrará aquella solución que mejores resultados proporcione en la fase de desarrollo.

3. **Módulo de clasificación de polaridad:** este módulo usa los datos recopilados por el módulo de adquisición e integración de información para clasificar la polaridad de la opinión correspondiente, según sea positiva, neutra o negativa.

4. **Módulo de clasificación de temática:** este módulo, al igual que el módulo de clasificación de polaridad, usa los datos recopilados por el módulo de adquisición de integración de información para clasificar o identificar la temática abordada en el texto analizado.

5. **Módulo de identificación de perfiles de opinión (*clustering*):** este módulo será el encargado de identificar el perfil de opinión de cada usuario. Mediante un algoritmo de *clustering* se está utilizando información contextual y de perfil de cada usuario para detectar de forma desasistida los distintos colectivos de opinión o *clusters*, de acuerdo a las similitudes encontradas en los datos.

La arquitectura final no depende del sector de ventas del usuario y, por lo tanto, puede ser extrapolada a cualquier sector de manera sencilla. Este hecho, da una versatilidad importante a nuestra herramienta cuyo uso no quedará restringido a ningún sector en particular.

Hito 2: Análisis de texto libre

El objetivo principal de este hito es entrenar y validar diferentes algoritmos de *Machine Learning* que permitan el análisis de texto libre. Este hito se divide en dos tareas independientes:

- T2.1: Construcción de corpus de sentimientos
- T2.2: Clasificación de polaridad de texto libre
- T2.3: Clasificación de temática de texto libre
- T2.4: Evaluación formal del analizador de texto libre

T2.1: Construcción de corpus de sentimientos

En esta tarea se han analizado distintos corpus de polaridad necesarios para análisis de sentimientos llevados a cabo a lo largo del proyecto AIDOS. A continuación, se describen brevemente cada una de estas fuentes de corpus de polaridad consultadas:

1. iSOL

iSOL es un corpus que ha sido desarrollado por el grupo de investigación SINAI de la Universidad de Jaén (http://sinai.ujaen.es/isol/).

El archivo correspondiente al corpus de polaridad positiva está formado por un total de 2509 términos, mientras que en el caso del corpus negativo, consta de 5626 entradas distintas.

2. Sentiment Lexicons in Spanish

Sentiment Lexicons in Spanish ha sido desarrollado por miembros del grupo *Language and Information Technologies* de *University of North Texas*.

Dicho corpus de sentimientos consta de dos ficheros de texto. Uno de ellos, contiene un lexicón más robusto, debido a las anotaciones manuales llevadas a cabo, mientras que con el segundo se procedió mediante anotaciones automáticas. Ambos casos tienen los términos positivos y negativos en el mismo archivo, etiquetados mediante "pos" o "neg" según corresponda.

El primero de los corpus consta de un total de 1347 términos, mientras que el segundo está formado por 2496 palabras.

3. SPANISH EMOTION LEXICON

Spanish Emotion Lexicon (http://www.cic.ipn.mx/~sidorov/) es un corpus de sentimientos donde, en lugar de etiquetar cada término con su polaridad (positivo-negativo), lo hace mediante distintos estados: *Alegría, Enojo, Miedo, Repulsión, Sorpresa y Tristeza.* En este caso, de modo que se pueda llevar a cabo un análisis de polaridad, se procede a identificar como positivo el estado *Alegría*, y como negativos el resto.

Spanish Emotion Lexicon consta de 2036 palabras con su estado asociado. Tras la diferenciación en positivo y negativo previamente explicada, ésta se resume en 668 palabras positivas y 1368 términos negativos.

4. ELHPOLAR

ElhPolar (http://komunitatea.elhuyar.eus/ig/files/2013/10/ElhPolar_esV1.lex) ha sido creado a partir de la traducción automática de un lexicón de polaridad inglés al español. Además, se ha llevado a cabo una anotación manual del lexicón resultante para mejorar la precisión de éste.

ElhPolar está compuesto por términos o pequeñas estructuras con su correspondiente polaridad. En total, nos encontramos con 3302 términos negativos y 1897 positivos. Además, del total de 5199 términos, 473 corresponden a estructuras de más de una palabra (*a_la_deriva, falta_de_respeto*).

Comparación experimental de corpus

Para llevar a cabo una comparación de los distintos corpus anteriormente descritos ha sido necesario disponer de un conjunto de textos previamente etiquetados con su respectiva polaridad, de modo que se puedan comparar los resultados obtenidos con los valores dados inicialmente.

Los datos utilizados constan de un total de 7219 mensajes de la red social de *microblogging Twitter* (comúnmente denominados *tweets*). Estos mensajes están todos escritos en español, con su etiqueta correspondiente a la polaridad. Dicha etiqueta puede tomar seis valores distintos:

- **P+**: polaridad muy positiva.
- **P**: polaridad positiva.
- **NEU**: polaridad neutra.
- **NONE**: sin polaridad.

- **N**: polaridad negativa.
- **N+**: polaridad muy negativa.

Dichos mensajes se distribuyen respecto a su polaridad como sigue:

P+	P	NEU	NONE	N	N+
1652	1232	670	1483	1335	847

Si bien sería razonable proceder con dichas etiquetas, se ha decidido simplificarlas de modo que se pase de seis valores distintos a tres a razón del limitado tamaño de dicha base de mensajes, combinándolas del siguiente modo:

Polaridad	Nueva etiqueta	Etiquetas antiguas
Positiva	1	P+, P
Neutra	0	NEU, NONE
Negativa	-1	N, N+

De este modo, cada uno de los mensajes pasa a estar etiquetado con una de las tres nuevas etiquetas (1, 0, -1).

A modo de resumen, se han tenido en cuenta las palabras de negación (*no, ni, nunca, jamás, etc.*) al influenciar la polaridad de las siguientes palabras en el mensaje, así como la posibilidad de incluir o no *stemming* en el proceso.

T2.2: CLASIFICACIÓN DE POLARIDAD DE TEXTO LIBRE

El proceso de analizar y clasificar la opinión de un texto es complejo. Como consecuencia, podemos distinguir distintos pasos a incluir o considerar a la hora de llevarlo a cabo. En los siguientes apartados se desarrolla cada uno de los pasos que hemos abordado. Nótese que todo el análisis ha sido desarrollado con el software libre R.

1. TOKENIZACIÓN

La tokenización consiste en la división del texto en unidades más simples, usualmente palabras. De este modo, es posible analizar cada una de estas unidades de forma individual y clasificarla según se considere conveniente.

Este proceso incluye a su vez la eliminación de *stop words*, palabras cuya carga léxica es muy baja y a su vez aparecen en los textos con una frecuencia elevada (preposiciones, determinantes, etc.). Estas

palabras pueden introducir ruido al analizar el texto, de ahí que resulte necesario eliminarlas para el posterior análisis.

2. STEMMING

El proceso de *stemming* se basa en la eliminación de las derivaciones léxicas de las palabras. El principal beneficio es la fusión de términos con una misma raíz, y que por tanto, tiene una carga semántica similar (*compra, comprar, comprados* tienen como raíz común *compr-*).

3. LEMATIZACIÓN

El proceso de lematización es muy similar al de *stemming* descrito en el anterior apartado. En este caso, en lugar de reducir una palabra a su raíz, se busca el lema correspondiente a dicho término.

El lema de una palabra es el término que representa todas las formas flexionadas de éste (véase un verbo y sus distintas conjugaciones).

4. N-GRAMAS

Un n-grama es una subsecuencia de n elementos dentro de una secuencia mayor. Debido a su mayor complejidad, la utilización de n-gramas (con n>1) supone un mayor coste computacional, aunque son ampliamente utilizados en la minería de textos al poder encontrar una fuerte correlación entre dos o más palabras, formando por ejemplo, una expresión común (*sin embargo, a pesar de*). Nótese que para n=1, coincide con el proceso de tokenización por palabras.

5. ANÁLISIS SINTÁCTICO

El análisis sintáctico de las estructuras de los textos puede permitir detectar distintos patrones que alteren el sentimiento y la polaridad de éstos. Estructuras de la forma "… pero …", "… sin embargo …" "A pesar de …, …", hacen que la polaridad en cada una de las partes influya en la otra.

Del mismo modo, existen modificaciones debidas a negaciones. Nótese que la polaridad de "la comida está bien" y "la comida no está bien" es totalmente opuesta por la inclusión del adverbio de negación "no". Para incluir la negación en el análisis de polaridad, hemos seleccionado las distintas palabras de negación en español, de modo que todos aquellos términos posteriores a éste con polaridad, hasta la siguiente parada de puntuación, verán su polaridad invertida.

Cabe destacar que el conjunto de mensajes de entrenamiento utilizado procede de la red social de *microblogging Twitter*, la cual tiene limitaciones de 140 caracteres por mensaje, por lo que apenas se encontrarán signos de puntuación que distingan varias frases dentro del mismo mensaje.

6. DETECCIÓN DE EMOTICONOS Y LÉXICO CON CARGA DE OPINIÓN

Los emoticonos son, hoy en día, una parte importante dentro de las diferentes redes sociales. Se utilizan de forma habitual para expresar emociones, o incluso reforzar el mensaje. La existencia de éstos, permite que se puedan generar corpus de mensajes a partir de ellos con una polaridad definida.

Es el caso de *Twitter*, donde se pueden llevar a cabo búsquedas de mensajes con emoticonos simples. Realizando una búsqueda con emoticonos positivos, se puede generar un corpus de tal polaridad. Del

mismo modo, se puede proceder con emoticonos con carga negativa para la construcción de un corpus negativo.

Sin embargo, también existe otro enfoque a la hora de aprovechar la existencia de los emoticonos en los mensajes de las redes sociales. El principal objetivo de éstos es mostrar una emoción asociada al mensaje, por lo que pueden ser de gran ayuda a la hora de determinar la polaridad de un mensaje. Incluyéndolos en los diccionarios léxicos, podrían utilizarse las codificaciones de los emoticonos como un término más que expresa polaridad positiva o negativa.

Hemos llevado a cabo tres experimentos principales. En el primero, hemos utilizado los corpus de polaridad. En el segundo, hemos utilizado métodos de aprendizaje automático con una regresión logística multinomial. Finalmente, hemos aplicado *Support Vector Machines* (SVM) en su versión multiclase, *One-vs-All*.

CORPUS DE POLARIDAD

El procesado aquí descrito tiene como pieza fundamental los corpus de polaridad. A continuación, pasaremos a describir los distintos pasos que hemos aplicado para dicho análisis:

- **Tokenización**: hemos aplicado la tokenización palabra a palabra, incluyendo como unidades también los símbolos de puntuación ".", "," y ";". Además, hemos eliminado las *stop words* del modo que se ha descrito en el apartado 2.1.
- ***Stemming***: hemos utilizado ambos enfoques, tanto incluyendo el proceso de *stemming*, como sin incluirlo.
- **n-gramas**: por extensión de la tokenización, hemos aplicado 1-gramas.
- **Análisis sintáctico**: se ha aplicado el procedimiento explicado anteriormente para incluir las negaciones como modificadores de la polaridad en las palabras posteriores a éstas hasta el siguiente símbolo de puntuación.

Como se resaltó con anterioridad, el corpus con mejores resultados ha sido *ElhPolar* en ambos casos. Además hemos visto que incluir el proceso de *stemming* nos lleva a la mejora de los resultados en la mayoría de las situaciones.

REGRESIÓN LOGÍSTICA MULTINOMIAL

En este caso, hemos aplicado una regresión logística multinomial. Para ello, los datos han sido divido en dos, 80% de los datos para entrenamiento, y 20% para test. Este proceso se ha llevado a cabo mediante el paquete *glmnet* del software R (https://cran.r-project.org/web/packages/glmnet/glmnet.pdf), mediante la función *cv.glmnet(…,family = 'multinomial',…)*.

En este caso, los pasos seguidos han sido los siguientes:

- **Tokenización:** hemos eliminado las *stop words*. El resto del proceso de tokenización se detalla en el apartado de n-gramas.
- ***Stemming***: al no utilizar corpus de polaridad, no se ha incluido el proceso de *stemming* por no ser tan necesario como en la experimentación con los corpus de polaridad.

- **n-gramas**: hemos aplicado dos versiones. Por un lado, 1-gramas, es decir, tokenización palabra a palabra, y por otro, 2-gramas, en busca de estructuras de dos palabras con una fuerte relación.
- **Análisis sintáctico**: al tratarse de un método con aprendizaje automático, no ha sido necesario incluir ningún proceso de análisis sintáctico.

SVM (One-vs-All)

Otro de los métodos aplicados está basado en las conocidas *Support Vector Machine* (SVM). Este método de clasificación binaria propone etiquetar los individuos con la correspondiente clase mediante una división de éstos en dos espacios mediante un hiperplano.

Sin embargo, las SVM están diseñadas, como se apuntaba previamente, para clasificación binaria. Por lo tanto, se ha recurrido al modelo *One-vs-All*, el cual permite aplicarlo a casos multiclase. La base del modelo *One-vs-All* radica en la clasificación de cada una de las clases (*One*) contra el resto de clases (*All*).

En este caso, los pasos seguidos han sido los siguientes:

- **Tokenización:** hemos aplicado la tokenización palabra a palabra. Además, hemos eliminado las *stop words* del modo que se ha descrito anteriormente.
- ***Stemming***: se ha incluido el proceso de *stemming* para que el cálculo de la medida TF-IDF que se explica a continuación tenga mayor consistencia.
- **n-gramas**: por extensión de la tokenización, hemos aplicado 1-gramas.
- **Análisis sintáctico**: al tratarse de un método con aprendizaje automático, no ha sido necesario incluir ningún proceso de análisis sintáctico.

Para poder llevar a cabo dicho proceso, se ha creado una matriz de términos de documentos, donde cada fila representa un documento, y cada columna una de las palabras que aparecen en todos ellos. Se trata de una matriz *sparse* (una matriz de grandes dimensiones donde casi todos los valores son 0).

El modelo SVM (*One-vs-All*) aplicado procede del paquete *e1071* del software R (https://cran.r-project.org/web/packages/e1071/e1071.pdf), mediante la función *svm*. En nuestro caso, se ha seleccionado un 80% de los datos como conjunto de entrenamiento, y un 20% como conjunto test.

Otros métodos analizados

Además de los mostrados en los apartados anteriores, también se han estudiado otros métodos que han sido descartados por proporcionar resultados insuficientes.

- **Modelo de Naïve-Bayes**: el modelo de Naïve-Bayes es un clasificador probabilístico que tiene como base el teorema de Bayes. Como base de dicho modelo, se asume que las variables predictivas son independientes entre sí.
- **Árboles de decisión**: a partir de reglas básicas se construyen los árboles de decisión, donde cada uno de los nodos finales, llamados comúnmente hojas, asigna a aquellos individuos que satisfacen tales condiciones una clase.

- **Polaridad de Turney**: se predice la polaridad en base a la orientación semántica, mediante la distancia (que debe ser predefinida) a los términos *excellent* y *poor* en inglés, o sus respectivas traducciones, *excelente* y *malo/mala*, en español.

Debe tenerse en cuenta que los resultados de los métodos desarrollados deben ser comparados con cautela, ya que el primero (*Corpus de polaridad*) disponen de todo el conjunto de mensajes para la evaluación, mientras que los restantes (*Regresión logística multinomial* y *SVM (One-vs-All)*) requieren de un conjunto de entrenamiento, lo que ve reducida la cantidad de mensajes en la fase de test.

T2.3: CLASIFICACIÓN DE TEMÁTICA DE TEXTO LIBRE

Con el objetivo de reconocer automáticamente la temática de un determinado texto a partir de un diccionario léxico, lo primero que se ha hecho es construir un corpus de expresiones lingüísticas específico para cada una de las temáticas que pretendemos identificar. En determinadas aplicaciones, este corpus de expresiones viene dado por la propia aplicación, o es conocido de antemano. Por ejemplo, si queremos rastrear cuando se habla de una determinada entidad (marca, producto, empresa), conocemos expresiones referenciales, como son los nombres propios, principalmente. En otras aplicaciones, el corpus de expresiones no es conocido de antemano y no se puede construir manualmente, bien por la complejidad que supondría hacerlo de esta manera o bien por desconocimiento de las expresiones más características de la temática en cuestión.

Para aquellas aplicaciones en las que es necesario construir el corpus de expresiones característico de cada temática, se ha desarrollado una aplicación que se encarga de rastrear un conjunto de documentos (textos relacionados con la temática) que permitan la extracción de las palabras utilizadas con mayor frecuencia.

Debido al gran volumen de opinión que genera de manera continua la red social *Twitter* en innumerables temáticas y sectores de opinión, se ha trabajado con la API gratuita que proporciona (https://dev.twitter.com/rest/public). Esta API, en su versión gratuita, presenta una serie de limitaciones, como por ejemplo, no permitir recoger más de 9 días de *tweets* desde el día de la consulta. Aun así, supone una opción muy interesante para captar mensajes de diferentes temáticas y analizar el vocabulario propio más utilizado para cada una de ellas.

A continuación se listan las cuentas de *Twitter* seleccionadas para construir el corpus de expresiones de cada temática:

Temática: **Política**

'marianorajoy', 'sanchezcastejon', 'Pablo_Iglesias_', 'agarzon', 'rosadiezglez', 'Albert_Rivera', 'EsperanzAguirre', 'ccifuentes', 'cayo_lara', 'GLlamazares', 'manuelacarmena', 'EduMadina', 'RevillaMiguelA', 'masaenfurecida', 'gobiernoespa', 'desdelamoncloa', 'Senadoesp', 'elpais_espana', 'ElMundoEspana'

Temática: **Cine**

'nosgustacine', 'horasperdidas', 'EPcine', 'CINEMANIA_ES', 'SensaCine', 'cineralia', 'fotogramas_es', 'Academiadecine', 'blogdecine', 'cineycine', 'Claquetados', 'Cinemaficionado'

Temática: **Tecnología**

'elpais_tec', 'abc_tecnologia', '_Ciencia_Tecno', 'xataka', 'tecnologiafacil', 'elmundoes_tecno', 'RedTecnologia', 'Omicrono', 'error500', 'marlexsystems'

Temática: **Moda**

'1sillaxamibolso', 'bymyheels', 'VB_victorblanco', 'galagonzalez', 'princepelayo', 'balamoda','carmeron', 'atrendylife', 'marialeonstyle', 'ladyaddict', 'armarioenruinas', 'stradivarius', 'Oysho', 'creadoresACME', 'MBFWMadrid', 'VogueSpain', 'GlamourSpain', 'marieclaire_es', 'VanityFairSpain', 'elle_es', 'cosmopolitan_es', 'telva', 'TrendysMx'

Temática: **Motor**

'marcamotor', 'AS_Motor', 'A_Motor', 'autopista_es', 'diariomotor', 'autofacil', 'motorpuntoes', 'car_and_driver', 'Motor16', 'km77com', 'periodismomotor'

Temática: **Deporte**

'diarioas', 'marca', 'mundodeportivo', 'sport', 'abc_deportes', 'elpais_deportes', 'ElMundoDeportes', 'losmanolostv', 'tjcope'

La elección de estas cuentas de *Twitter* se debe a que son cuentas específicas de cada temática, por lo tanto, la descarga de un gran volumen de comentarios y opiniones proporciona una buena aproximación de la terminología que se emplea en cada disciplina.

El modo de proceder ha sido el siguiente:

a) Descarga de todos los *tweets* que la API de *Twitter* permite relacionados con cada una de las temáticas seleccionadas.
b) Filtrado de cada uno de los *tweets*, con el objetivo de quedarse con aquellas palabras con interés semántico. Para ello, el proceso consiste en el uso de minúsculas y la eliminación de caracteres extraños, de acentos y de *stop words*.
c) Almacenamiento de las palabras resultantes en una "bolsa" de palabras utilizadas en la temática.
d) Análisis estadístico por frecuencia de aparición de cada una de las palabras resultantes, de manera que las más frecuentes pasen a formar parte del corpus de expresiones de cada temática.

A modo de ejemplo se muestran la nube de palabras (representación visual de las palabras que conforman un texto, donde el tamaño es mayor para las palabras que aparecen con mayor frecuencia) para una de las temáticas analizadas:

Temática: **Política**

Figura 3. Nube de palabras correspondientes a la temática "Política"

Cabe destacar que el corpus de expresiones utilizadas más frecuentemente en cada temática es dinámico y muy dependiente del momento en que se construye, razón por la cual hay que repetir este proceso cada cierto tiempo. Así, por ejemplo, si ocurre un determinado evento, es probable que aparezcan expresiones relacionadas con ciudades, personajes, etc. entre las expresiones más utilizadas, pero que tienen un carácter temporal y muy asociado al evento en concreto, no a la temática en general.

Una vez que tenemos un corpus de expresiones características de cada una de las temáticas que queremos identificar, se ha desarrollado un algoritmo que clasifica automáticamente la temática del texto.

La generación del algoritmo que permite la clasificación automática de la temática se dividió en dos fases: entrenamiento y validación.

T2.4: EVALUACIÓN FORMAL DEL ANALIZADOR DE TEXTO LIBRE

Las métricas que se han aplicado para la comparativa de resultados han sido el porcentaje de acierto o *accuracy*, la precisión y el *recall*.

Siendo $v_1 = -1, v_2 = 0, v_3 = 1$, y $M = (m_{ij})_{i,j=1,2,3}$ la matriz de confusión 3x3, donde m_{ij} representa el número de casos con valor real v_i y estimado como v_j para $i = 1,2,3,\ j = 1,2,3$:

- ***Accuracy***:

$$\frac{\sum_{i=1}^{3} m_{ii}}{\sum_{i=1}^{3}\sum_{j=1}^{3} m_{ij}}$$

- **Precisión**:

$$\frac{1}{3}\sum_{i=1}^{3}\frac{m_{ii}}{\sum_{j=1}^{3} m_{ij}}$$

- ***Recall***:

$$\frac{1}{3}\sum_{i=1}^{3}\frac{m_{ii}}{\sum_{j=1}^{3} m_{ji}}$$

La primera de estas métricas, representa el porcentaje de acierto del método con respecto al valor que toma la etiqueta de polaridad dada inicialmente. La precisión por otro lado, muestra la proporción de falsos positivos, mientras que el *recall* lo hace con la proporción de falsos negativos.

Cabe destacar que dichas métricas, cuanto más cercanas a 1, mejor actuación del modelo de clasificación, significando que existe un porcentaje de acierto muy alto (*accuracy*), una proporción de falsos positivos muy baja (precisión) y una proporción de falsos negativos muy baja (*recall*).

1. CLASIFICADOR DE OPINIÓN

Se han analizado distintas variantes del clasificador de opinión. Se ha procedido a su comparación, que será desarrollada en mayor detalle a lo largo de este apartado.

Corpus de polaridad

El primero de los métodos aplicados ha sido la utilización del corpus de polaridad para clasificar la opinión de un texto. Hemos analizado distintos corpus de polaridad para clasificar la opinión de los mensajes. Estos corpus se listan a continuación:

- *iSOL*
- *Sentiment Lexicons in Spanish (Robust)*
- *Sentiment Lexicons in Spanish (Medium)*
- *Spanish Emotion Lexicon*
- *ElhPolar*

También se ha tenido en cuenta la posibilidad de incluir un proceso de *stemming*, por lo que en total, se han considerado 10 combinaciones distintas ([5 corpus] x [Sí/No *stemming*]).

Regresión logística multinomial

El segundo método aplicado para clasificar opinión está centrado en la regresión logística multinomial. Para ello, los datos han sido divididos en entrenamiento (80%) y test (20%). En este caso, se han distinguido dos factores a la hora de determinar el modelo. Por un lado, hemos tenido en cuenta la posibilidad de utilizar 1-gramas y 2-gramas. Además, se han aplicado dos metodologías distintas para la representación de documentos mediante matrices *sparse*. Dichos métodos son *Vocabulary-based vectorization y Feature hashing*. En total, se han analizado cuatro combinaciones distintas.

SVM (One-vs-All)

El tercer método aplicado es la extensión a multiclase de las *Support Vector Machines* (SVMs), comúnmente llamado *One-vs-All*. En este caso, se enfrenta cada una de las clases al resto. De nuevo, se ha llevado a cabo una división del conjunto en entrenamiento (80%) y test (20%), aplicándole al primero *cross-validation* con 5 iteraciones.

2. CLASIFICADOR DE TEMÁTICA

Se han analizado distintas variantes del clasificador de temática. Mediante la experimentación se ha procedido a su comparación, que será desarrollada en mayor detalle a lo largo de este apartado.

Regresión logística multinomial

Al igual que ocurría con el método de clasificación de polaridad, el segundo método aplicado para clasificar la temática está centrado en la regresión logística multinomial. Para ello, los datos han sido divididos en entrenamiento (80%) y test (20%).

Hito 3: Identificador de perfiles de opinión

El objetivo principal de este hito es desarrollar un algoritmo de *clustering* que sea capaz de recoger toda la información recopilada y analizada durante los Hitos 1 y 2, con el propósito de identificar distintos perfiles de opinión. Esto permitirá adoptar medidas de marketing personalizadas, centradas en tipos de cliente específicos, analizando aquellos *clusters* o grupos de usuarios más interesantes a la hora de plantear una campaña de promoción, así como determinar qué tipo de comunicación adoptar con respecto a cada tipología de cliente.

En estos momentos se están empezando a diseñar los primeros experimentos que permitan detectar de manera automatizada los distintos grupos de usuarios. Por el momento se está planteando un *clustering* jerárquico, basado en el autodescubrimiento de *clusters* en base a jerarquías. Estas jerarquías se representan habitualmente por medio de dendogramas, en los cuales a partir del conjunto de individuos se van formando grupos progresivamente en base a la distancia existente entre ambos (distancia euclidiana, Manhattan, etc.).

Por el momento no existen resultados relevantes para presentar en este documento.

Hito 4: Deep Learning

Como ya se ha comentado en varias ocasiones a lo largo de este documento, buena parte de la información desestructurada existente se encuentra en formato de audio. En estos momentos se están empezando a evaluar las principales herramientas que permiten realizar lo que se conoce como "*speech to text*". En particular, se está realizando un estudio detallado de las APIs más conocidas, como son *Google Speech Recognition, Microsoft Bing Voice Recognition* o *IBM Speech to Text*. Además de las anteriores, se está explorando la opción de desarrollar un módulo propio de "*Speech to Text*", basado en la herramienta Sphinx.

ANUALIDAD 2017

Hito 3: Identificador de perfiles de opinión

Este Hito se ha desarrollado entre los meses de diciembre de 2016 y abril de 2017. El objetivo principal ha sido desarrollar un algoritmo de *clustering* capaz de recoger toda la información recopilada y analizada durante los Hitos 1 y 2, con el propósito de identificar distintos perfiles de opinión. Esto permitirá adoptar medidas de marketing personalizadas, centradas en tipos de cliente específicos, analizando aquellos *clústeres* o grupos de usuarios más interesantes a la hora de plantear una campaña de promoción, así como determinar qué tipo de comunicación adoptar con respecto a cada tipología de cliente.

La actividad se ha desglosado en las siguientes tareas:

- T3.1: Representación de un modelo de usuario y marca.
- T3.2: Análisis de datos de opinión para prospectiva de Mercado.

T3.1: REPRESENTACIÓN DE UN MODELO DE USUARIO Y MARCA

El objetivo de esta tarea es la representación de un modelo "usuario-marca" para lo cual se ha tomado como base la información que proporciona el analizador de texto desarrollado y validado en el Hito 2, a la que se ha añadido información relativa al usuario y al contexto en el que se ha producido la conversación.

Para analizar la información contextual y de perfil de usuario a adicionar a la información que proporciona el analizador, ha sido necesario conectarse a la API oficial de la red social para poder analizar los campos que ofrece.

A partir de esta API, se pueden analizar los mensajes donde se menciona una palabra o una cuenta oficial concreta. Estos mensajes proporcionan información de contexto de la conversación, entre las cuales se han seleccionado las siguientes:

- *screenName*: nombre de la cuenta del usuario.
- *text*: texto del mensaje.
- *created*: fecha y hora de publicación del mensaje.
- *favoriteCount*: número de veces que ha sido marcado el mensaje como favorito hasta el momento de la descarga.
- *retweetCount*: número de veces que ha sido retuiteado el mensaje hasta el momento de la descarga.

A partir del screenName de estos mensajes, se puede acceder a los perfiles de los usuarios que han mencionado la marca o cuenta, obteniéndose así toda la información que proporciona la API sobre dichas cuentas (mediante el comando lookupUsers (cuentas).

De todas los datos proporcionados, se considera que los que pueden ser de utilidad para su posterior utilización son:

- *screenName*: nombre de la cuenta del usuario.
- *description*: texto libre donde cada usuario puede escribir una breve descripción propia.
- *statusesCount*: número de mensajes publicados.
- *followersCount*: número de seguidores del usuario.
- *favoritesCount*: número de favoritos del usuario.
- *friendsCount*: número de cuentas que sigue el usuario.
- *url*: URL opcional que puede añadir a su perfil el usuario.
- *name*: nombre del usuario.
- *created*: fecha de creación de la cuenta.
- *protected*: indicador de protección de la cuenta.

Para la construcción del modelo usuario-marca, se han eliminado las variables que no serán relevantes en el proceso de *clustering* por razones obvias: *description*, *url*, *protected*, *name*, *created* y *favoriteCount*.

Por otro lado, ha sido necesario seleccionar qué enfoque aplicar para el tratamiento de la temática y la franja de publicación, habiendo elegido finalmente aquéllos que incluyen toda la información en una sola variable. De este modo, los modelos tendrán la información asociada a la temática y la franja de publicación resumida en una variable, evitando problemas por el exceso de variables, conocido comúnmente como *curse of dimensionality*.

T3.2: ANÁLISIS DE DATOS DE OPINIÓN PARA PROSPECTIVA DE MERCADO

Esta tarea persigue el desarrollo de un algoritmo de *clústering* para la identificación de perfiles de usuarios en función de sus opiniones respecto a una marca previamente estipulada.

Para alcanzar este objetivo, se han diseñado experimentos que detectan de manera automatizada los distintos grupos de usuarios. Se ha planteado un *clustering* jerárquico, basado en el autodescubrimiento de *clústeres* en base a jerarquías, representadas habitualmente por medio de dendogramas, en los cuales a partir del conjunto de individuos se van formando grupos progresivamente en base a la distancia existente entre ambos (distancia euclidiana, Manhattan, etc.).

Para ello, se ha procedido en un primer lugar, a descargar toda la información necesaria de dicha red social. En concreto, los mensajes que mencionan a la marca, la información de cada uno de los usuarios que profieren dichas opiniones, así como las *timelines* de dichos usuarios.

A partir de dicha información, se ha realizado un análisis de texto libre basado en los desarrollos del Hito 2 de este proyecto. En concreto, se ha analizado la polaridad de los mensajes que mencionan a la marca, clasificando las *timelines* de cada usuario en función de unas temáticas predeterminadas que caractericen sus gustos, así como la frecuencia de publicación de cada usuario con respecto al momento del día en el que es más activo en dicha red social.

Una vez generadas todas las variables adicionales, se procede a la selección de atributos relevantes mediante un análisis descriptivo de los datos, así como un análisis de correlaciones que permitan

identificar variables redundantes. A partir de dicha selección, se procede con el particionado o *clustering*. En concreto, se han aplicado tres enfoques distintos, *Partitioning Around Medioids* (PAM), *clustering* jerárquico y *mixture models* para datos mixtos.

Finalmente, la presentación de los resultados se ha hecho mediante textos que incluyan dicha información de un modo más interpretable. Además, se ha proporcionado un resumen de los datos asociados a cada uno de los conjuntos generados por cada método de *clustering* aplicado.

Se detallan a continuación los pasos del algoritmo de *clústering* desarrollado:

1.- Descarga de mensajes: Los únicos parámetros de entrada del algoritmo que debe proporcionar el usuario son el nombre de la marca a analizar y su cuenta oficial en Twitter. Es necesario disponer de ambas, de modo que se puedan recabar mensajes de los usuarios dirigidos tanto a la cuenta oficial, como nombrando la marca sin necesidad de haber mencionado la cuenta. Para proceder a descargar los mensajes de Twitter, ha sido necesario conectarse a la API oficial de dicha red social. Para ello, se ha registrado una cuenta como desarrollador, y se han obtenido las claves de acceso. Para dicha conexión con la API a través de R, se ha utilizado el paquete twitteR, diseñado para tal fin.

2.- Análisis de texto libre: En esta segunda fase del proceso se tiene como piedra angular la aplicación de los distintos desarrollos llevados a cabo en el Hito 2 de este proyecto.

3.- Selección de atributos: Una vez descargados los mensajes y añadidas las variables procedentes del análisis de texto libre, se ha procedido a seleccionar los atributos de cara al análisis de las posibles metodologías de clustering sobre dichos usuarios.

En este caso, solo será necesario trabajar con la información de las cuentas de los usuarios que han mencionado a la marca en cuestión, al estar toda la información relevante recogida de sus menciones y *timelines* en las variables obtenidas como se explica en el apartado anterior.

4.- Clustering: Existen diversas técnicas de clustering en la literatura. En este caso se ha tenido en cuenta que nos encontramos con datos mixtos, es decir, cuantitativos (influencia, polaridad, *statusesCount*) y cualitativos (franja, temática), por lo que ha sido necesario adaptar dichos procedimientos y desarrollar una metodología específica adecuada a las particularidades del caso.

Hito 4: Deep Learning

Este Hito se ha desarrollado entre los meses de noviembre de 2016 y agosto de 2017. Su objetivo ha sido analizar las principales técnicas de *Deep Learning* y desarrollar e implementar un algoritmo *"Speech to Text"*. Para llevar alcanzar este objetivo se ha desarrollado la tarea "T4.1: Análisis e implementación de las principales herramientas "Speech to Text".

T4.1: Análisis e implementación de las principales herramientas "Speech to Text"

Las técnicas de *Deep Learning* comprenden aquellos algoritmos que intentan reproducir la mecánica del cerebro humano en la codificación y decodificación de mensajes y permiten el auto aprendizaje. Se corresponden con una evolución de las clásicas redes neuronales. Actualmente, su utilización ha cobrado gran relevancia en los campos de tratamiento de la imagen, procesado del lenguaje y tecnologías del

habla. En este hito se han analizado y evaluado diferentes herramientas del ámbito *del Deep Learning*, que permiten realizar lo que se conoce como "*speech to text*", teniendo en cuenta que una parte muy importante de la información desestructurada que existe en la actualidad se encuentra en formato de audio.

En primer lugar, se han analizado los tipos de topologías de redes neuronales más utilizados bajo la denominación de Deep Learning, que son las redes neuronales convolucionales (convnets), las redes neuronales recurrentes (RNNs), los autoencoders y las Restricted Boltzmann Machines (RBMs).

Las redes neuronales convolucionales suelen utilizarse en tareas de visión artificial. A partir de ellas se han conseguido grandes avances en los últimos tiempos en detección y reconocimiento de objetos, aunque también han sido utilizadas con éxito para tareas de reconocimiento de habla; incluso se ha probado experimentalmente su aplicación en sistemas de recomendación (Spotify).

Las redes neuronales recurrentes son las más utilizadas para tareas relacionadas con el procesamiento de lenguaje natural, como reconocimiento de habla o traducción automática. Una variante de este tipo de redes son las Long Short Term Memory (LSTM). Además, ejemplos de combinaciones de una arquitectura mixta basada en una red neuronal convolucional y una RNN (o LSTM) se utilizan para la descripción automática de imágenes.

Las RBMs y los autoencoders destacan como técnicas fundamentales para el aprendizaje no supervisado. La concatenación de varias unidades de RBMs se conoce como Deep Belief Network (DBN) y es un método muy eficaz para la extracción de características relevantes y de alto nivel para la representación de datos, especialmente de cara a la generación y reconocimiento de imágenes, aunque sus ámbitos de aplicación son diversos.

Otras herramientas que disponen de unas funcionalidades bastante completas, y donde se pueden encontrar ejemplos enfocados en aspectos más concretos o topologías específicas son cuda-convnet, OverFeat o cuDNN, centradas en la implementación de convnets; o nolearn, que implementa también DBNs. También existen suites de Machine Learning que permiten hacer un uso rápido de resultados y topologías de Deep Learning de una forma sencilla con un enfoque de mucho más alto nivel, como el que encontramos en GraphLab Create o en Azure ML.

Posteriormente se ha estudidado el estudio del uso del Deep Learning en aplicaciones de Procesamiento del Lenguaje Natural.

En los últimos años, los algoritmos de Deep Learning han sido la clave para obtener un salto muy significativo en las prestaciones de los sistemas de reconocimiento automático del habla. Las redes neuronales han demostrado ser una herramienta versátil capaz de modelar todos los aspectos acústicos, fonéticos y lingüísticos asociados con esta tarea. Los complejos sistemas tradicionales basados en una multitud de componentes específicos han sido ya sustituidos por estructuras genéricas de gran versatilidad y mayores prestaciones.

El Procesamiento del Lenguaje Natural (NLP) es una disciplina con una larga trayectoria. Nace en la década de 1960, como un sub-área de la Inteligencia Artificial y la Lingüística, con el objeto de estudiar los problemas derivados de la generación y comprensión automática del lenguaje natural.

El objetivo de los sistemas automáticos de Procesamiento de Lenguaje Natural es el de desarrollar sistemas inteligentes capaces de comprender el lenguaje verbal humano (oral y escrito) aportando utilidad al usuario.

Principalmente, un modelo de procesamiento de lenguaje natural se divide en una arquitectura de niveles que permite analizar el texto obteniendo de él, diferentes características según el nivel de abstracción que se estudie. La arquitectura (ordenada de manera ascendente) se estructura normalmente en cinco niveles de análisis: nivel fonológico, nivel morfológico, nivel sintáctico, nivel semántico y nivel pragmático.

El reconocimiento automático del habla es una disciplina de la inteligencia artificial que tiene como objetivo permitir la comunicación hablada entre seres humanos y computadoras. El problema que se plantea en un sistema de este tipo es el de hacer cooperar un conjunto de informaciones que provienen de diversas fuentes de conocimiento (acústica, fonética, fonológica, léxica, sintáctica, semántica y pragmática), en presencia de ambigüedades, incertidumbres y errores inevitables para llegar a obtener una interpretación aceptable del mensaje acústico recibido.

Un sistema de reconocimiento de voz es una herramienta computacional capaz de procesar la señal de voz emitida por el ser humano y reconocer la información contenida en ésta, convirtiéndola en texto o emitiendo órdenes que actúan sobre un proceso determinado, como es el caso de los chatbots.

Llegados a este punto, se han analizado las herramientas más destacadas en conversión de audio a texto:

- El "Cloud Speech API" de Google, que permite a los desarrolladores la conversión de audio a texto aplicando potentes modelos de redes neuronales de forma sencilla utilizando su entorno de desarrollo. Es capaz de reconocer hasta 80 idiomas y sus variantes.

- El módulo "*Speech to Text*" de IBM: Dentro del catálogo de servicios disponibles en Bluemix, IBM ha creado una serie de servicios cognitivos que permiten enriquecer de forma sencilla las aplicaciones. Dentro de estos servicios hay disponibles APIs para hacer análisis de sentimientos de textos, convertir voz a texto y viceversa, o extraer información de fotografías, entre muchos otros.

- Y el "*HPE Haven On Demand*" de HP: Plataforma de inteligencia artificial que facilita APIs de aprendizaje automático y servicios que permite a desarrolladores y empresas crear aplicaciones móviles o empresariales data-rich. HPE Haven OnDemand se integra dentro de los servicios de Microsoft Azure y ofrece más de 60 APIs y servicios de análisis de aprendizaje sobre una amplia gama de datos, incluyendo texto, audio, imagen, aplicaciones sociales, web y video. Dentro de esta plataforma, destaca la Speech Recognition API, la cual es capaz de crear transcripciones de texto a partir de ficheros de audio en 16 idiomas distintos.

Hito 5: Integración y validación del sistema en el sector de la alimentación

Este Hito se ha desarrollado entre los meses de septiembre y diciembre de 2017. Su objetivo ha sido la implementación correcta de la API de Twitter en el sistema desarrollado y descarga automática de

Tweets con menciones a las entidades seleccionadas en el sector de la alimentación. Para ello, se han desarrollado las siguientes tareas:

- T5.1: Integración de los componentes software.
- T5.2: Construcción de corpus de entidades para el sector de la alimentación.
- T5.3: Validación de resultados en el sector de la alimentación.

T5.1: INTEGRACIÓN DE LOS COMPONENTES SOFTWARE.

Esta tarea tiene como objetivo la obtención de una herramienta que permita analizar el sector de la alimentación a través de la red social Twitter. Para ello se integrarán los componentes software desarrollados a lo largo de los trabajos ejecutados en el *Hito 2: Análisis de texto libre*, en el *Hito 3: Identificador de Perfiles de opinión*, y en el *Hito 4: Deep Learning*.

Dicha información es el punto de partida del análisis posterior. Para tal fin, se han utilizado los desarrollos del Hito 2 asociados al análisis de polaridad.

Posteriormente, se clasifican los distintos usuarios. Para ello, se ha utilizado la herramienta desarrollada para tal fin en el Hito 2.

De un modo similar, se ha desarrollado un clasificador de tipología del producto, donde se ha adaptado el clasificador del Hito 2 previamente mencionado a un caso más particular como el que aquí se presenta. Así, será posible detectar si algún tipo de producto recibe una mayor cantidad de mensajes positivos o negativos por parte de los usuarios para una determinada marca. Nótese que se ha considerado utilizar información recogida de otras fuentes distintas a Twitter mediante el uso de la herramienta de Speech to Text desarrollada en el Hito 4, si bien la información obtenida mediante la API de Twitter ha sido de mayor utilidad de cara a la generación de dicho corpus.

T5.2: CONSTRUCCIÓN DE CORPUS DE ENTIDADES PARA EL SECTOR DE LA ALIMENTACIÓN.

A lo largo de esta tarea se ha construido un corpus de expresiones lingüísticas específico para la temática que pretendemos identificar, en este caso es el sector alimentación. Para ello, siguiendo el procedimiento desarrollado en anteriores tareas, se ha trabajado con la red social *Twitter*, en concreto con la API gratuita que proporciona. Esta API, en su versión gratuita, supone una opción muy interesante para captar mensajes y analizar el vocabulario propio más utilizado. Como contrapartida, presenta una serie de limitaciones, la principal de las cuales es el no permitir recoger más de nueve días de *tweets* previos al día de la consulta.

Para la generación del corpus de forma automática, se ha seleccionado el sector de la alimentación como temática y dentro de ésta, se ha escogido un conjunto de cuentas populares de *Twitter* que generan opinión sobre ellas.

Cabe destacar que el corpus de expresiones utilizadas más frecuentemente en una temática es dinámico y muy dependiente del momento en que se construye, razón por la cual ha de repetirse este proceso cada cierto tiempo. Así, por ejemplo, si ocurre un determinado evento, es probable que aparezcan expresiones relacionadas con ciudades, personajes, etc. entre las expresiones más utilizadas, pero que tienen un carácter temporal y muy asociado al evento en concreto, no a la temática en general.

Por otra parte y para complementar al corpus anterior, se ha generado de manera semiautomática un nuevo corpus perteneciente al mismo sector. Para su construcción se han utilizado fuentes de conocimiento, bases de datos y catálogos de productos de proveedores del sector de la alimentación. Con esta información, se ha realizó un pre-procesamiento inicial de los datos, agrupando éstos en siete grandes clases, como se explica a continuación.

A partir de un conjunto de productos del sector de la alimentación organizados y clasificados por el proveedor en familias, subfamilias, sector y nombre de producto, se creó un grupo de clases nuevas para clasificar.

T5.3: VALIDACIÓN DE RESULTADOS EN EL SECTOR DE LA ALIMENTACIÓN.

En esta tarea se han aplicado los distintos desarrollos llevados a cabo a lo largo del proyecto a un caso particular: el sector de la alimentación.

El objetivo ha sido validar las distintas herramientas desarrolladas en hitos anteriores y adaptándolas al caso de experimentación en el sector seleccionado, para lo cual se han estudiado distintas características, tales como la polaridad de los mensajes, las temáticas generales o el tipo de producto del que habla un usuario al mencionar a la marca.

Para esta validación se ha analizado cada una de las marcas contemplando:

- Datos sobre la descarga.
- Análisis de polaridad. A partir del módulo de polaridad desarrollado en el Hito 2, se asocia a cada una de las menciones la polaridad de dicho mensaje. Se realizará un resumen acerca de la polaridad de dichos mensajes, mostrando la proporción de menciones etiquetadas como positivas, neutras y negativas.
- Clasificador de temática general. A lo largo del Hito 2 de este proyecto se ha desarrollado un clasificador de temática, donde se etiquetan los mensajes en seis categorías distintas:
    - Deporte.
    - Cine.
    - Moda.
    - Tecnología.
    - Motor.
    - Política.
- Clasificador de tipología del producto. Se ha generado un lexicón con distintas tipologías de producto del sector de la alimentación.
- Franja horaria de publicación. Como se considera en el Hito 3, se ha recogido para cada usuario la hora de publicación de los mensajes en su *timeline*.

- Análisis de influencia.
- Datos de *clustering*. Se proporcionará un resumen de las particiones obtenidas mediante los tres métodos tenidos en cuenta en el Hito 3.

Se reflejan a continuación los resultados para una de las marcas analizadas según estos criterios:

- Datos sobre la descarga.

Tabla con un breve resumen de los datos descargados.

Menciones descargadas	5.662
Cuentas analizadas	4.437
Mensajes descargados de *timelines*	938.144

Tabla 1: Tabla resumen de los datos sobre la descarga

- Análisis de polaridad.

Tabla de frecuencias de polaridad de las cuentas con respecto a sus menciones:

Polaridad	Frecuencia
Negativa	1.801
Neutra	1.189
Positiva	1.447

Tabla 2: Tabla resumen del análisis de polaridad

En la **Figura 2** se muestran en forma porcentual dichos datos representados mediante un gráfico de sectores.

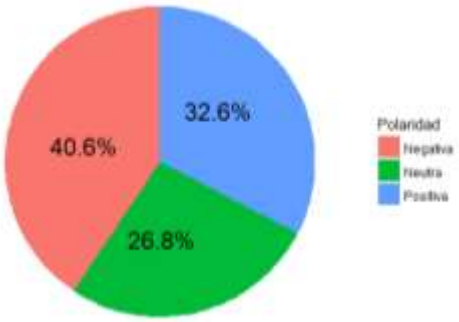

Figura 2. Frecuencia de polaridad

- Clasificador de temática general.

Tabla de frecuencias de temáticas localizas en las *timelines* de los usuarios que han mencionado, mostrando tanto las generales en la primera columna de datos, como particularizando a cada uno de los valores de la polaridad.

Temática \ Polaridad	General	Negativa	Neutra	Positiva
Cine	202	81	57	64
Deportes	2.662	1.060	725	877
Moda	813	327	206	280
Motor	7	2	4	1
Política	673	311	169	193
Tecnología	66	16	24	26

Tabla 3: Tabla resumen del clasificador de temática general

En la **Figura 3** se muestra, mediante un diagrama de barras, el porcentaje (eje Y) de cuentas localizadas para cada una de las temáticas (eje X), con respecto a la polaridad (leyenda), es decir, el caso general y las tres posibles etiquetas de polaridad.

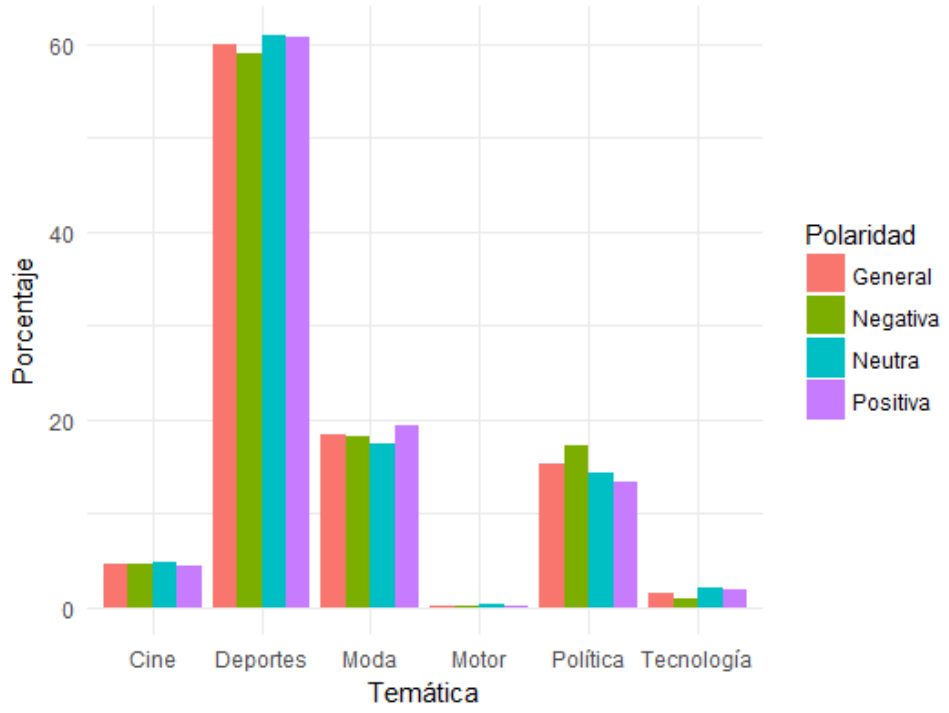

Figura 3. Porcentaje de aparición de cada temática con respecto a la polaridad

Como se puede observar, las proporciones de cada una de las temáticas son similares para los cuatro casos en estudio.

- Clasificador de tipología del producto.

Tabla de frecuencias de tipología de producto localizadas en las menciones, mostrando tanto las generales en la primera columna de datos, como particularizando a cada uno de los valores de la polaridad.

Tipología de producto \ Polaridad	General	Negativa	Neutra	Positiva
Bebidas	269	115	89	65
Charcutería – carnicería – pescadería	211	56	46	109
Conservas – congelados	238	104	69	65
Frutas – hortalizas	52	29	11	12
Productos lácteos	264	131	46	87
Perfumería – higiene	443	187	122	134
Otros	686	265	161	260

Tabla 4: Tabla resumen de los resultados del clasificador de tipología del producto

En la **Figura 4** se muestra, mediante un diagrama de barras, el porcentaje (eje Y) de cuentas localizadas para cada una de las tipologías de producto (eje X), con respecto a la polaridad (leyenda), es decir, el caso general y las tres posibles etiquetas de polaridad.

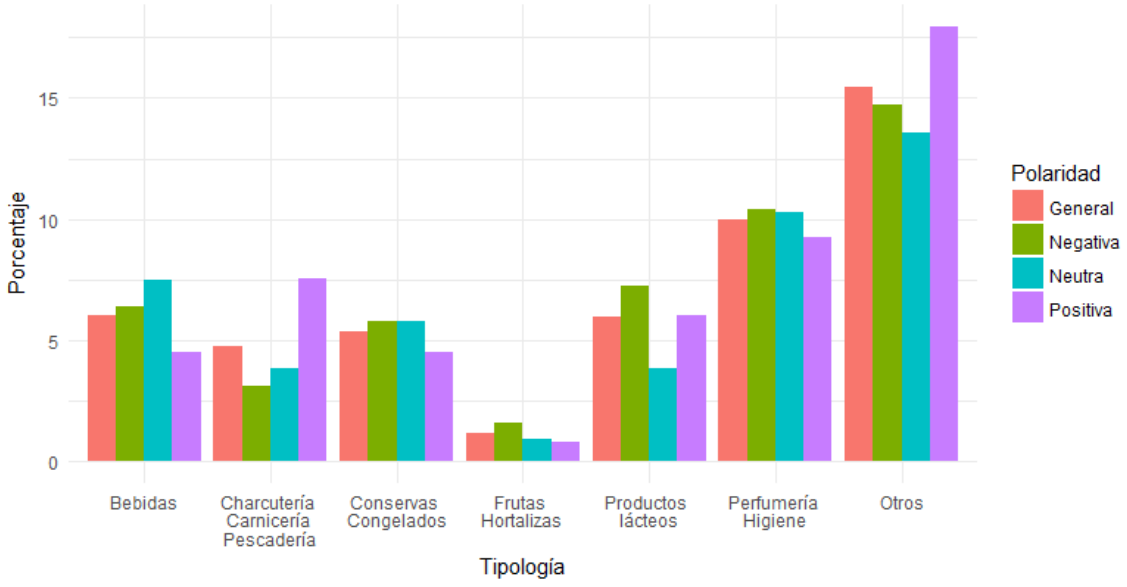

Figura 4. Porcentaje de aparición de cada tipología con respecto a la polaridad

A diferencia del caso de la temática, sí que se pueden encontrar diferencias en los comentarios a la marca entre polaridad positiva y negativa. En concreto, se puede observar como los productos de la clase "Charcutería – carnicería – pescadería" tienen una mayor proporción de mensajes positivos que negativos, así como en la categoría "Otros". También se observa cierta tendencia hacia mensajes negativos en la clase "Bebidas".

- Franja horaria de publicación.

Tabla de frecuencias de franja horaria de publicación localizadas en las *timelines* de los usuarios que mencionan, mostrando tanto las generales en la primera columna de datos, como particularizando a cada uno de los valores de la polaridad.

Franja horaria \ Polaridad	General	Negativa	Neutra	Positiva
Mañana	1.960	766	549	645
Tarde	1.817	751	474	592
Noche	651	281	163	207

Tabla 5: Tabla resumen del análisis de la franja horaria de publicación

En la **Figura 5** se muestra, mediante un diagrama de barras, el porcentaje (eje Y) de cuentas localizadas para cada una de las franjas de publicación (eje X), con respecto a la polaridad (leyenda), es decir, el caso general y las tres posibles etiquetas de polaridad.

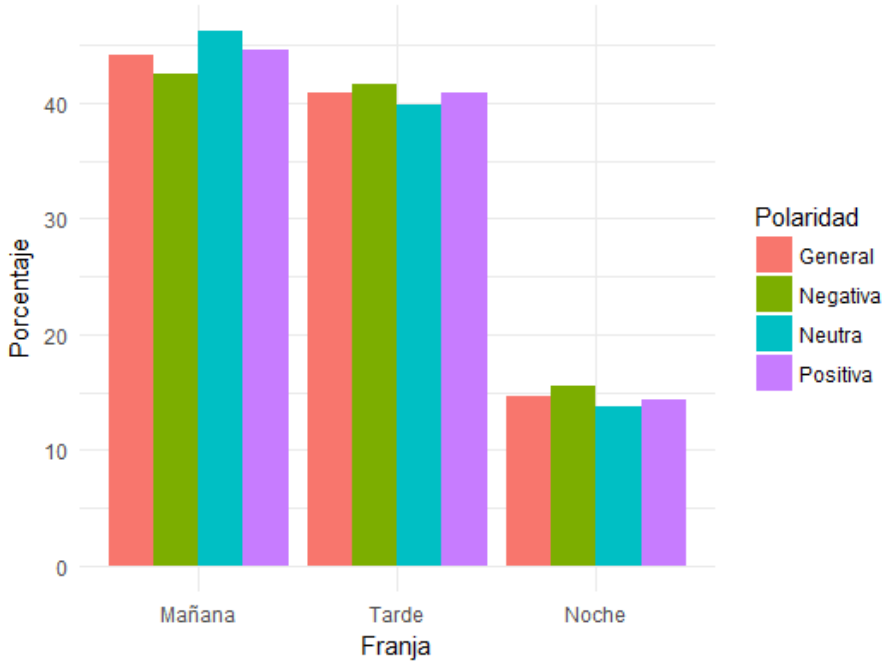

Figura 5. Porcentaje de franja del día de uso con respecto a la polaridad

En este caso, se puede observar como no existen diferencias significativas en la franja de publicación con respecto a la polaridad de las menciones.

- Análisis de influencia.

Tabla resumen de datos asociados a la medida de influencia generada. Se proporcionan distintos datos estadísticos de dicha variable que permiten comparar el caso general y los casos condicionados a la polaridad de sus menciones.

Valor \ Polaridad	General	Negativa	Neutra	Positiva
Mínimo	-4,16	-4,1	-4,16	-3,26
Primer cuartil	-0,61	-0,62	-0,66	-0,53
Mediana	0,07	0,05	0,08	0,09
Media	0,21	0,16	0,17	0,29
Tercer cuartil	0,72	0,69	0,7	0,79
Máximo	10,34	9	10,34	8,85
Desviación típica	1,36	1,33	1,36	1,4

Tabla 6: Tabla resumen del resultado del análisis de influencia

En la **Figura 6** se muestran los datos expresados en la tabla previa, mediante diagramas de cajas o *boxplots*. Cada caja representa la influencia (eje Y) con respecto a la polaridad (eje X), es decir, el caso general y las tres posibles etiquetas de polaridad.

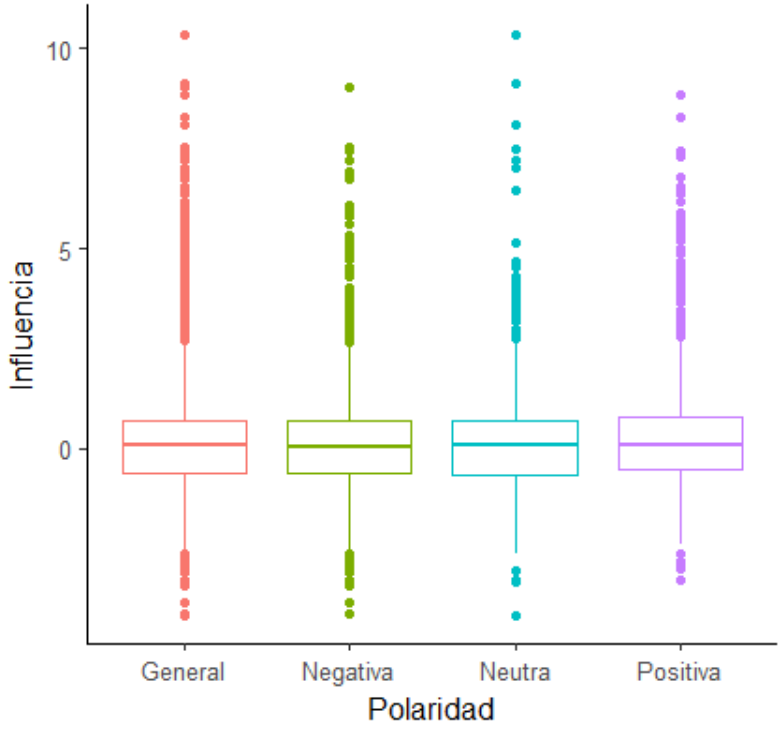

Figura 6. *Boxplots* representativos de las distribuciones de la variable influencia

Como se puede observar tanto en la tabla como en la figura, la influencia en los cuatro casos analizados en función de la polaridad es similar, no existiendo ninguna diferencia significativa entre ellos.

- Datos de clustering.

La siguiente tabla muestra el número de *clusters* generados por cada uno de los tres métodos aplicados a los datos.

Método	*Clusters*
Partitioning Around Medoids	3
***Clustering* jerárquico**	8
***Mixture models* para datos mixtos**	11

Tabla 7: Tabla resumen de los datos de *clustering*

No es posible llevar a cabo una representación gráfica del último de los modelos, por lo que procedemos a mostrar los resultados del método *Partitioning Around Medoids* (**Figura 7**) y el *clustering* jerárquico (**Figura 8**).

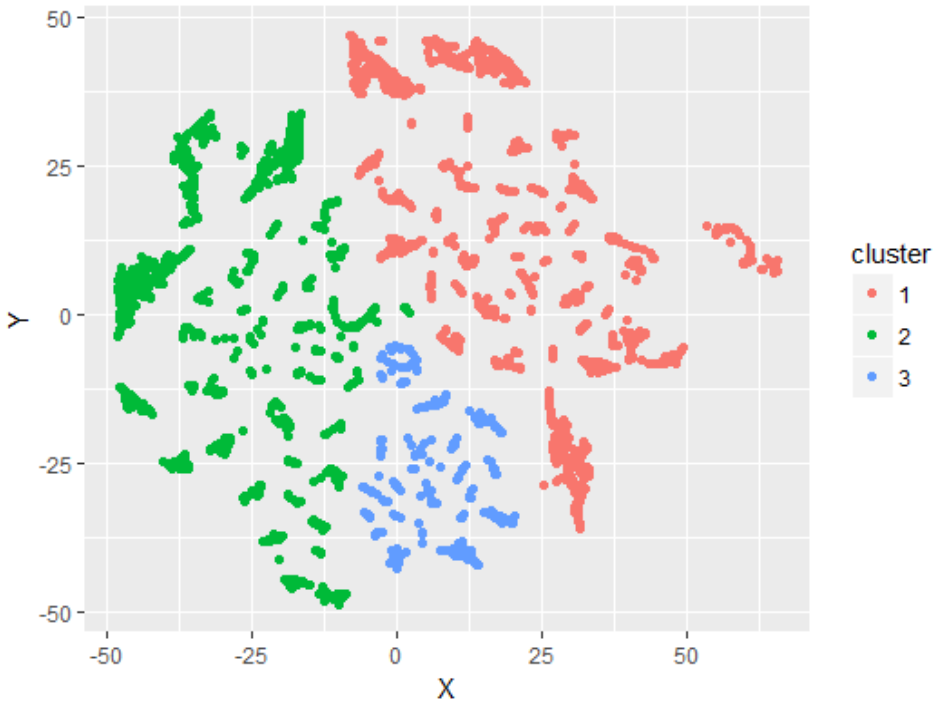

Figura 7. *Clustering* obtenido mediante método PAM de 3 conjuntos

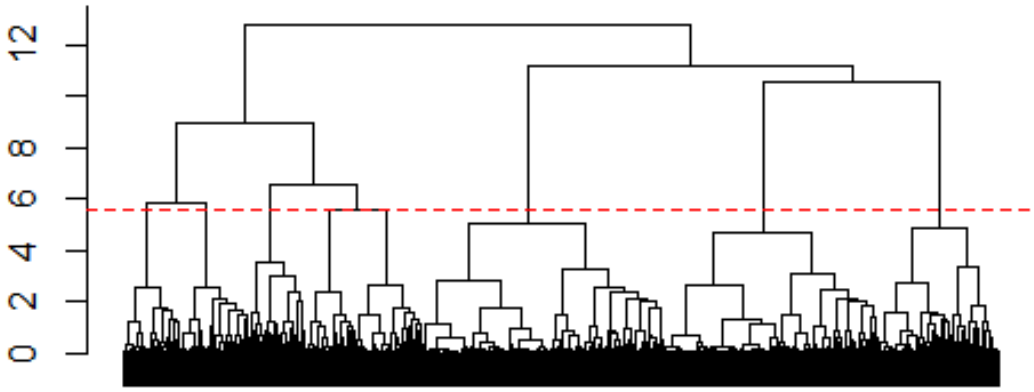

Figura 8. *Clustering* jerárquico de 8 conjuntos

2. RESULTADOS CONSEGUIDOS

Durante la ejecución del proyecto se han conseguido todos los resultados planificados, tal y como se había planteado al inicio del proyecto y que se detallan a continuación, relacionándolos con los objetivos específicos del proyecto:

- Diseño e implementación de un sistema capaz de recoger y analizar información desestructurada desde diferentes fuentes de información.
- Extracción de conocimiento relativo al análisis de texto libre, basado en las opiniones y sentimientos de los usuarios respecto a una entidad determinada (empresa, producto o marca).
- Identificación automática de los distintos perfiles de opinión que existen en torno a una entidad.
- Transcripción automática a texto de aquellas fuentes de información desestructurada que se presentan en formato de audio.
- Aplicación de las técnicas generadas para el análisis de una red de comunicación de microblogging.