laSalle

UNIVERSITAT RAMON LLULL

Escola Tècnica Superior d'Enginyeria La Salle

Treball Final de Màster

Màster Universitari en Enginyeria Informàtica i la seva gestió

Análisis y diseño de videoclipping

Alumne
David Turpín Franco

Professor Ponent
Maria Antonio Mozota Coloma

ACTA DE L'EXAMEN DEL TREBALL FI DE CARRERA

Reunit el Tribunal qualificador en el dia de la data, l'alumne

D. David Turpín Franco

va exposar el seu Treball de Fi de Carrera, el qual va tractar sobre el tema següent:

Análisis y diseño de videoclipping

Acabada l'exposició i contestades per part de l'alumne les objeccions formulades pels Srs. membres del tribunal, aquest valorà l'esmentat Treball amb la qualificació de

Barcelona,

VOCAL DEL TRIBUNAL VOCAL DEL TRIBUNAL

PRESIDENT DEL TRIBUNAL

Abstract

Análisis y diseño general de una aplicación basada en el uso de técnicas de Information Retrieval para la obtención de nuevo conocimiento en base al análisis de información no estructurada originaria de un repositorio de vídeos online. A partir de unos parámetros iniciales de búsqueda, el objetivo final de la aplicación es la obtención de un vídeo con extractos de instantes relevantes para el usuario de otros contenidos multimedia.

Realización de un documento de especificación de requerimientos para el detalle de las funcionalidades de los módulos que forman la aplicación. Análisis de los requisitos y diseño general de la aplicación en base al estudio de la interacción con la API de datos de YouTube.

Analysis and general design of an application based on the usage of Information Retrieval techniques for the acquisition of new knowledge coming from unstructured information allocated in online media repositories. Through a definition of initial search parameters, the goal of the application is to get a new video made up from extracts of the relevant results for the user.

Development of a Software Requirements Specification Document to get the detailed functionalities of different modules that compose the application. Requirements and general design analysis of the application based on the study of the interaction with the YouTube API.

Resumen

El objetivo de este proyecto es realizar un estudio de la viabilidad de implementación de una aplicación para el análisis de información asociada a videos disponibles a través de Internet.

En este proyecto se elabora un análisis y diseño general de la aplicación, que se enmarca dentro del área de *Web Mining*, para tratar de obtener conocimiento a partir de datos no estructurados y disponibles en un repositorio de videos online.

En base a unos parámetros de búsqueda iniciales, y haciendo uso de un conjunto de técnicas de *Information Retrieval*, esta aplicación efectúa una búsqueda y un posterior análisis sintáctico de los contenidos asociados a los videos para determinar la relevancia de cada uno de ellos. Como resultado de la ejecución se obtiene un nuevo video con extractos de los videos relevantes derivados de la consulta, en aquellos instantes en que se han mencionado los términos de búsqueda o palabras relacionadas.

En el plano teórico, y para poder determinar las técnicas a utilizar durante el diseño de la aplicación, se realiza un estudio de las diferentes técnicas de minería de datos en el entorno web. Del mismo modo, se estudia el conjunto de herramientas disponible en el marco de *Information Retrieval* para el procesamiento de documentos, útiles para el tratamiento de la información a analizar asociada a los videos: los comentarios y los subtítulos.

Por otro lado, se realiza un pequeño estudio de los diferentes repositorios de video *online* disponibles en Internet para determinar sobre cuál de ellos ejecutar el proceso de búsqueda y análisis, y se elabora un estudio en profundidad de la metodología de interacción con el portal seleccionado.

A continuación se desarrolla un análisis y diseño general del prototipo de la aplicación, redactando un documento de especificación de requerimientos basado en la norma *IEE 830-1998 "IEEE Recommended Practice for Software Requirements Specifications"* en el que se detalla el conjunto de módulos que forman la aplicación, así como sus entradas, salidas y funcionalidades.

Con respecto al análisis de requerimientos y al diseño general, se describe el esquema de funcionamiento general de la aplicación, así como el funcionamiento de cada uno de los módulos que la forman, detallando para cada uno de ellos los casos de uso relativos a todas las funcionalidades definidas en el documentos de especificación de requerimientos.

Finalmente, se realiza un diseño de la arquitectura física necesaria para la ejecución de la aplicación propuesta, teniendo en cuenta la selección de las tecnologías necesarias a desarrollar para cada uno de los elementos que la forman.

Índice

1 Introducción

1.1 Marco

El volumen de información disponible hoy en día en Internet es cada vez mayor y más diverso. Gran cantidad de información se ofrece al usuario a pocos clics de ratón de distancia, y hoy en día el reto no está en saber si un determinado dato existe o no, sino en saberlo encontrar.

Por norma general, la información de que se dispone no está organizada: no es fácil de encontrar un origen fiable o con referencias, se fragmenta en diferentes fuentes e incluso puede estar disponible a través de distintos medios que pueden ser tanto escritos como audiovisuales. La información es cambiante, los datos se actualizan a gran velocidad y es difícil tener una trazabilidad de los cambios para tener una visión actualizada de la realidad.

En referencia a los medios, los datos en Internet ya no son sólo páginas web estáticas de contenido fijo: además de artículos hay wikis, donde los usuarios pueden renovar el contenido de forma autónoma; hay blogs, actualizados periódicamente por los mismos usuarios, y de los que pueden derivar comentarios; hay foros, preguntas con respuestas cruzadas que generan comunidades de usuarios de intereses comunes; y por supuesto hay contenido multimedia, gran cantidad de elementos como imágenes, audio o vídeos repletos de información.

Con el cambio (o evolución) de los contenidos de Internet, cambian también las pautas de navegación. Si hace un tiempo acceder a un buscador como Google para obtener información era el primer paso de todo usuario, no es extraño ahora encontrar motores de búsqueda de videos *online* como YouTube o DailyMotion como página inicial de una búsqueda, en especial en las nuevas generaciones de internautas.

Así pues la información se fragmenta, se divide entre diferentes fuentes y distintas formas de visualización, y se propaga por una comunidad de usuarios que no deja de crecer. Los nuevos usuarios aportan nueva información, actualizan contenidos y participan de la comunidad, incrementando así la necesidad de herramientas de gestión de un volumen de datos que no deja de crecer.

En este contexto, las técnicas de Recuperación de la Información (*Information Retrieval*[1]) tienen un papel fundamental en la recuperación y limpieza de los contenidos que la web nos puede ofrecer. Una vez recuperados los datos, que en el marco de este proyecto provienen de un entorno web, es necesario aplicar un conjunto de técnicas de Minería de Datos (en este caso minería Web), para poder extraer conocimiento útil que sea utilizado por nuevas herramientas que muestren y organicen la información de una forma distinta. En este proyecto esta aplicación recibe el nombre de VideoClipping.

La nueva organización de la información puede proporcionar nuevas formas de conocimiento. Bien sea al mostrar información específica por dominio o por actividad, o bien sea por ofrecer al usuario nuevas formas de consulta y visualización de la información anteriormente desestructurada. Estas herramientas pueden resultar de gran ayuda tanto en la gestión de los datos ya existentes como en la obtención de la información de una forma diferente a la conseguida hasta la fecha, de modo que pueda aportar valor al usuario final.

[1] En adelante IR (de *Information Retrieval*)

1.2 Objetivos

Los objetivos de este proyecto se enmarcan dentro de la realización de un proyecto web de tratamiento de contenidos audiovisuales y de información asociada a los mismos. Con este objetivo, se realiza un estudio de las diferentes técnicas de procesado de la información anteriormente mencionadas, con el fin de obtener una nueva forma de datos que pueda ser procesada de forma distinta a la fuente de datos original, aportando un nuevo valor al usuario final.

El proyecto se centra en aquellos elementos que puedan estar relacionados con los videos *online* por ser contenidos de gran difusión en los últimos tiempos y carecer de una estructura de la información clara, así como de herramientas de gestión adecuadas para el proceso interno de este tipo de contenido.

De este modo, los objetivos a tratar durante la realización de este proyecto son los siguientes:

- **Diseño de una herramienta de extracción y gestión de información de contenido audiovisual:**
  Realizar el diseño general del sistema de extracción y procesado de información asociada a un repositorio de videos *online*: VideoClipping. Realizar el diseño siguiendo estándares de definición de requerimientos para alinearse con unas prácticas adecuadas en lo que a metodología del software se refiere.

- **Estudio del marco teórico de las diferentes técnicas de Minería de Datos en un entorno web:**
  Analizar los diferentes elementos que componen la Minería de datos Web y describir cada uno de los elementos, prestando especial atención a aquellos que puedan estar relacionados de alguna forma con el contenido Audiovisual actualmente disponible en los principales repositorios de videos *online*.

- **Estudio de las principales técnicas de Recuperación de la Información (*Information Retrieval*) en un entorno web:**
  Estudiar las diferentes técnicas de IR aplicables a los contenidos presentes en el web, analizando las posibilidades de extracción de información en este tipo de entorno, así como aspectos relativos a la evaluación de la información extraída.

2 Web Mining: Data Mining en entorno web

2.1 Introducción al Web Mining

Las cifras de tráfico y usuarios de internet crecen a pasos agigantados. Según datos de Cisco hechos públicos recientemente[2], durante los próximos años se mantendrá esta tendencia e incluso irá al alza en alguna región, junto con un incremento en la generación datos de todo tipo y el consumo de contenidos audiovisuales, tanto a través del ordenador como de otros dispositivos.

Este escenario proporciona un jugoso repositorio para la minería de datos en el entorno web, al disponer no sólo de datos relativos a la información en sí misma, sino de métricas y referencias relativas al uso de la información: quién usa qué, cuándo y para qué.

Buscar, comprender y ser capaz de utilizar la gran cantidad de información disponible hoy en día en este entorno resulta un importante reto para el usuario medio. En ocasiones no es posible disponer de la información deseada a tiempo, puesto que esta es dinámica y está mal estructurada. Por ello, ya hoy en día los buscadores hacen uso de diversas técnicas de Minería de Datos basadas en web para optimizar sus búsquedas y ofrecer a los usuarios términos relevantes en cada momento.

Web Mining se puede definir como un conjunto de técnicas orientadas al descubrimiento y el análisis de la información útil alojada en Internet a través del estudio de un conjunto de datos no estructurado para obtener conocimiento desconocido hasta la fecha. Estos datos iniciales pueden estar alojados en diferentes fuentes: en un servidor que hospeda una página web, en el propio navegador del cliente, en un servidor Proxy (que hace de puente temporal entre un grupo de usuarios y un sitio web) o directamente en una base de datos.

Por lo tanto, la recuperación y el análisis de los datos depende de varios factores que condicionan las posibles técnicas a utilizar: dónde se encuentran los datos y qué tipo de disponibilidad tienen (por ejemplo, si son estáticos o se cargan de forma dinámica a través de una Base de Datos), qué tipo de datos son (si se trata de texto, de registros de acceso o de otros elementos), o incluso del tipo de usuarios del que se ha extraído la información (segmento de la población u otras características que pudieran condicionar los datos de entrada).

El origen de *Web Mining* proviene de la Minería de Datos clásica, que se originó por la necesidad de entender los hábitos de consumo de los clientes estudiando la “cesta de la compra”. Muchas técnicas de minería de datos se basan en este objetivo, como las Reglas de Asociación, la Clasificación o el *Clústering*.

2.2 Datos, Información y Conocimiento

Antes de entrar en el detalle de los diferentes elementos que forman el Web Mining, conviene distinguir entre Datos, Información y Conocimiento.

Como se ha mencionado anteriormente, a través de la web se acumula una gran cantidad de datos de los que no es posible extraer conocimiento de forma directa. Estos datos necesitan ser procesados, analizados y transformados mediante

[2] Estudio “Cisco Visual Networking Index: Forecast and Methodology, 2009-2014”

diferentes herramientas de extracción y análisis que permitan obtener información y de ella conocimiento.

Así, podemos distinguir estos tres elementos de la siguiente forma:

- **Datos:**

Se trata de la representación mínima de un atributo, un símbolo codificado (de forma numérica, alfabética, etc.) que describe las características de un elemento o una transacción o cualquier hecho real. Es la mínima unidad semántica, un conjunto de valores que por sí solos son irrelevantes, puesto que no argumentan el porqué de las cosas.

De forma genérica los datos pueden ser internos a la empresa, ser el resultado de transacciones o provenir de fuentes externas. En el entorno web estos datos pueden estar representados de distintas formas: texto plano, imágenes, contenido audiovisual, etc.

Los resultados de un análisis de *Web Mining* se basan en los datos, pero al reflejar sólo una parte de la realidad, nunca pueden ser tomados como elementos únicos, sino como orientativos acerca de la importancia o no de las cosas y de la interpretación de unos resultados.

- **Información:**

Es un conjunto de datos procesados que adquieren relevancia y que son de interés para el receptor del análisis de *Web Mining*. Se puede entender la información como el mensaje a transmitir a partir de un conjunto de datos de entrada, haciendo que se pueda cambiar la percepción inicial de los datos al dar forma a un gran volumen de entrada.

Esta extracción del volumen de datos de entrada se hace de un modo estructurado, y se organiza con el propósito de conseguir un tipo de resultados en particular. La transformación de estos datos en información puede hacerse mediante diversos métodos[3]:

- **Contextualizando**: Conociendo el contexto y el propósito para el cual se registraron o generaron los datos.
- **Categorizando**: Al conocer los componentes que forman los datos y poder dividirlos en categorías.
- **Calculando**: Aplicando operaciones matemáticas que ayuden a sintetizar el volumen de entrada para poder obtener conclusiones.
- **Corrigiendo**: Eliminando datos inconsistentes o errores que puedan distorsionar los resultados del análisis.
- **Condensando**: Resumiendo los datos de en grupos que agreguen diferentes conjuntos del volumen de datos de entrada.

[3] Según el esquema definido por Tomas H. Davenport y Larry Prusak del libro *Working Knowledge (1998)*.

De algún modo se puede entender la información como una forma de comunicación de los hechos que han sido extraídos a partir de los datos mediante las diferentes técnicas de Minería de Datos, y que pueden tener un impacto sobre el resultado final del análisis.

Teniendo en cuenta el contexto en que el que tienen valor los datos recogidos, que en el entorno web puede ser muy diverso, los propios datos y la utilidad de los mismos, y el tipo de transformación que se haga sobre los datos iniciales, la información recopilada será una u otra.

- **Conocimiento:**

La información se transforma en conocimiento cuando ésta es analizada y usada para algún tipo de proceso de toma de decisiones o para la realización de ciertas acciones tras el análisis de los datos de entrada. Por ello, el hecho de disponer de cierta información de un dominio concreto, avalada por la experiencia de personal cualificado capaz de tomar decisiones en función de estos resultados para resolver problemas complejos, es considerado conocimiento.

Así, el conocimiento es un intangible formado por diversos elementos como los valores, la información o experiencia previa, que permite el desarrollo de acciones y la toma de decisiones que aportarán nueva información. Por lo tanto no está sólo en los propios datos, sino en los mismos procesos de análisis de la información, en el procesado posterior que se haga y en las personas que lo apliquen.

Igual que la información deriva de los datos, los conocimientos surgen de la información. El conocimiento extraído de este modo puede venir ocasionado por la necesidad de solventar una situación específica, como respuesta a un problema, o bien de un modo pasivo tras el análisis con modelos matemáticos del volumen de entrada. En este sentido, muchas empresas han desarrollado en los últimos años sistemas internos para incrementar, almacenar y compartir su conocimiento, que es percibido como un bien intangible muy valorado.

La gestión del conocimiento difiere de la gestión de la información, si bien ambas disciplinas buscan desarrollar entornos que puedan ayudar a su personal en el proceso de toma de decisiones de problemas complejos. Y es que es importante observar que la gestión de la información trabaja con datos estructurados, cuantitativos y normalmente almacenados en bases de datos, mientras que la gestión del conocimiento trabaja con una entrada desestructurada, o en ocasiones implícita en los propios procesos de la actividad comercial de una empresa.

Teniendo en cuenta lo anterior, para poder extraer conocimiento es importante realizar un conjunto de acciones[4]:

- **Comparar con otros elementos**: Disponiendo de otro tipo de información que permita la comparación entre elementos de similares características.

- **Predicción de consecuencias**: A partir de experiencias previas o de la evolución del entorno poder intuir futuros comportamientos.

- **Búsqueda de conexiones**: Disponiendo de fuentes de información diversas ser capaz de relacionar los diferentes elementos entre sí.

[4] Según las acciones definidas por Tomas H. Davenport y Larry Prusak del libro *Working Knowledge (1998)*.

- **Conversación con otros portadores de conocimiento**: Estableciendo una comunicación con otras personas u organizaciones generadoras de conocimiento.

A modo de resumen, se puede establecer un gráfico de la relación entre estos tres elementos:

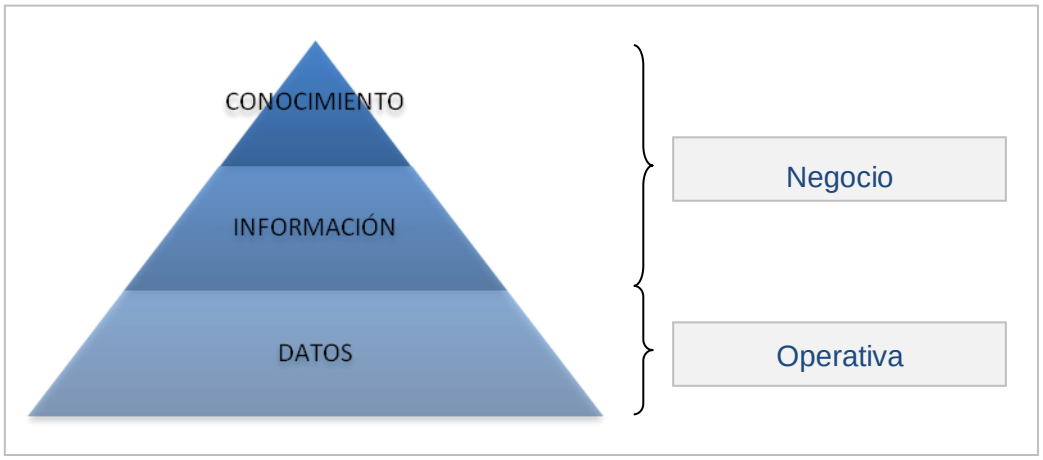


Ilustración 1: Conocimiento, Información y Datos. (Fuente: Davenport, T. y Prusak, L. 1998. Working Knowledge)

La recogida de datos es procesada por los elementos más sistemáticos y operativos, encargados de recopilar grandes volúmenes de datos, mientras que la recogida de información y la extracción de conocimientos se llevan a cabo por elementos correspondientes a la inteligencia de negocio.

2.3 Tipos de Web Mining

El tipo de datos que se pueden analizar en el entorno web es muy diverso, y en consecuencia las técnicas utilizadas para su análisis también son diferentes. En función del tipo de datos a analizar podemos distinguir diferentes técnicas que difieren de las que se podrían utilizar en la Minería de Datos convencional.

En el siguiente diagrama se muestra un esquema de los métodos de minería web y sus relaciones:

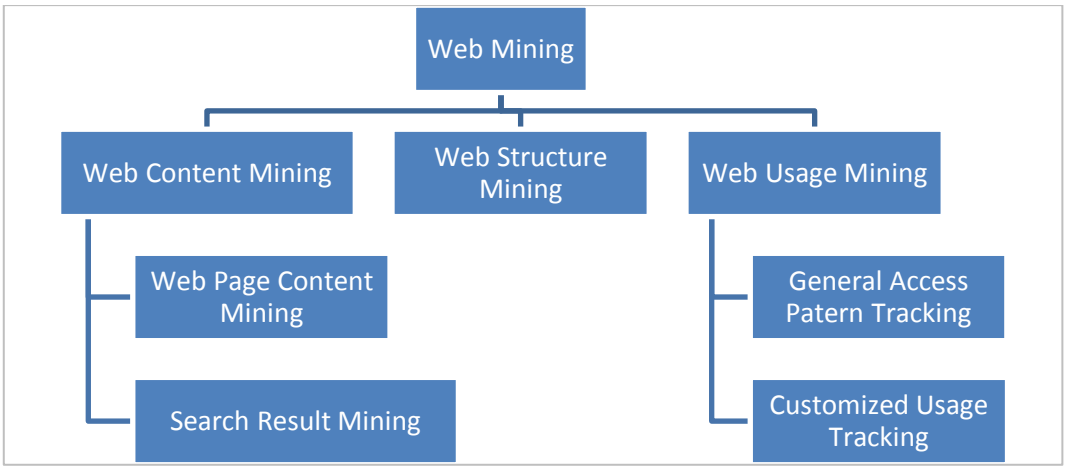


Ilustración 2: Esquema de tipos de Minería de Datos. (Fuente: Building an Intelligent web - Rajendra Akerkar, 2008)

Los datos de la web no son más que una gran colección de contenido, interacciones de usuario y registros de acceso que contienen ciertos patrones susceptibles de ser analizados y de extraer conocimiento nuevo, desconocido hasta la fecha, a partir de ellos.

En los siguientes apartados se explican de forma breve estos elementos. En próximos apartados de este proyecto se describe de forma detallada las técnicas utilizadas en el análisis de los contenidos web, ámbito en el que se centran los objetivos de este proyecto.

2.3.1 Uso web

El Web Mining orientado al Uso web es el estudio de los datos generados a partir de la navegación del usuario en las diferentes sesiones de navegación. Una sesión es el conjunto de páginas que un usuario visita dentro de un mismo sitio web, desde que entra hasta que sale de él o cierra el explorador.

Al navegar por internet se genera de forma indirecta un conjunto de datos que quedan almacenados a través de diferentes elementos y que trazan el comportamiento seguido durante la comunicación con la web. Esta información secundaria de la utilización de los recursos es almacenada a través de *Logs* de acceso a los diferentes portales a los que se navega.

El estudio de estos registros de acceso de los diferentes sitios puede ayudar a entender el comportamiento de los usuarios, así como la forma en que se estructura la información a través de las diferentes páginas visitadas.

Los datos recolectados a través del uso del web pueden ser tanto registros de acceso a sitios web como registros de los servidores proxy que den paso a los usuarios a los diferentes portales. Del mismo modo, se puede trazar y registrar también el comportamiento de los usuarios dentro de un mismo sitio web, generando un listado de los diferentes *clics* de ratón que el usuario realiza en el portal, con información relativa a la posición en la web y a los elementos afectados, entre otros.

Por lo tanto, se pueden registrar datos desde dos puntos de vista distintos:

- **Cliente**: La información recogida en el lado del cliente puede ayudar a entender aspectos de usabilidad de los sitios web al analizar la secuencia de *clics* y ser capaz de entender las posibles mejoras a realizar en la estructuración de los contenidos dentro de un sitio web.

- **Servidor**: Información relativa al flujo de usuarios de un sitio web a través de un servidor puede ayudar a entender los hábitos de consulta de los diferentes apartados de un sitio para mejorar la estructuración de los contenidos internos de un portal.

En el siguiente esquema, se muestra el proceso de minería del uso de un sitio web con el fin de encontrar patrones de comportamiento en los usuarios que puedan ayudar a mejorar la estructura de los datos:

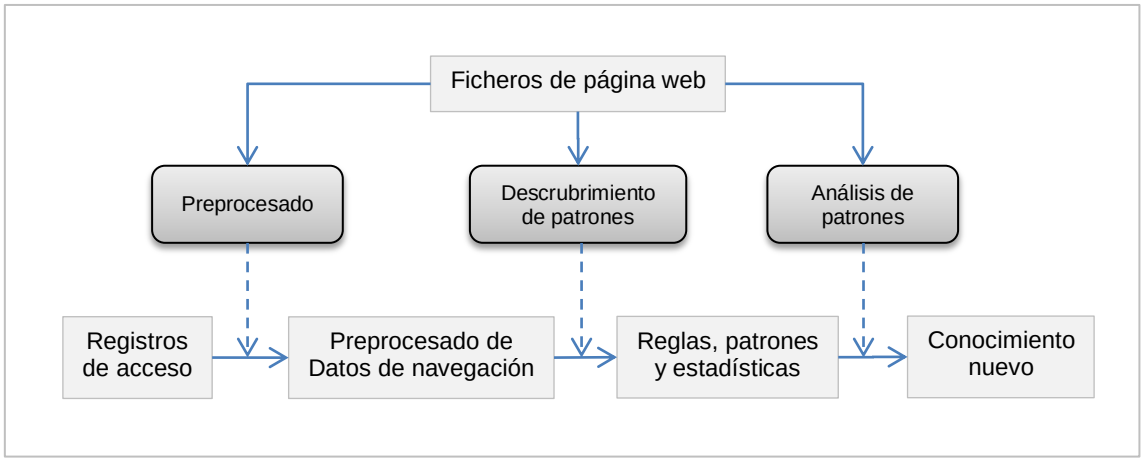


Ilustración 3: Proceso de Minería de Uso web. (Fuente: Building an Intelligent web - Rajendra Akerkar, 2008)

Los datos almacenados a través de los registros de acceso al sitio web han de ser procesados, filtrando la información relevante según el tipo de análisis. Posteriormente, hay que pre-procesar la información relativa a los *clic* de ratón efectuados por los usuarios, determinando por ejemplo la zona del sitio en la que sobre la que se ha actuado.

Una vez pre-procesados y limpiados los datos de entrada, se aplican las herramientas de Minería de Datos necesarias para poder obtener pautas de comportamiento entre los usuarios del sitio, obteniendo las estadísticas, reglas o patrones necesarios que aporten valor al usuario final de esta información (el administrador del sitio, el departamento de marketing, etc.)

Conviene destacar que el resultado final de un proceso de análisis de la utilización de un portal puede tener objetivos muy diversos, como pueden ser entre otros la personalización del sitio a las necesidades o características del usuario, la mejora del sistema de navegación o de gestión de la información del sitio, o bien un cambio en la estructura del portal, entre otros.

A modo de ejemplo, el uso de este tipo de técnicas se utiliza para predecir el grado de interés de un usuario con respecto a un apartado del sitio web, evaluando la secuencia de apartados que ha visitado éste mismo usuario y comparándola con la navegación que realizaron otros usuarios anteriormente. De este modo, se puede determinar la probabilidad de interés de un apartado para un usuario y modificar dinámicamente elementos de la web haciendo uso, por ejemplo, de algoritmos de clasificación como el C4.5.

2.3.2 Estructura web

Al hacer una consulta en un buscador web éste consulta los datos de que dispone en busca del término solicitado por el usuario. Sin embargo, este término no es suficiente para poder obtener un rango de valores adecuados o de calidad. Los recursos sobre los que se haga la búsqueda (en este caso páginas web) pueden no tener todo el contenido que necesita el usuario, o bien la información puede estar repetida, o ser de mala calidad. Por ello es necesario analizar la relación entre los diferentes recursos que contienen el término buscado para proporcionar al usuario los resultados de la consulta que le puedan resultar más relevantes.

Este tipo de técnicas tienen como objetivo analizar el entramado de enlaces que hay entre las páginas web, con el fin de poder extraer información a partir de las relaciones de las diferentes páginas. El estudio de la estructura de enlaces puede aportar información relativa al tipo de página, para buscar similitudes a su vez con otras páginas de características similares.

En relación a la búsqueda en función de la estructura de los sitios, hay algunos aspectos que conviene tener en cuenta y que fácilmente pueden darse en el mundo real:

- **Redundancia de información**: Es posible que más de un sitio web apunte a un mismo recurso, por lo que el resultado de la búsqueda sería redundante. Conviene evitar este tipo de situaciones con el estudio de las relaciones entre los diferentes resultados obtenidos.

- **Poco texto**: En ocasiones hay recursos que son fundamentalmente gráficos, con elementos visuales que no ayudan en un proceso de búsqueda a través de texto (como es el caso). En estos casos la información puede extraerse de las de páginas que apuntan a este recurso, y es esta estructura la que puede aportar información.

- **Diversos idiomas**: El contenido de las páginas puede estar, como es natural, en diferentes idiomas. Elementos de traducción serían útiles, pero ralentizarían la búsqueda del usuario, por lo que el estudio de las relaciones entre las diferentes páginas puede ayudar en este sentido.

- **Contenido irrelevante**: En ocasiones los desarrolladores añaden al final de las páginas o en apartados ocultos un conjunto de palabras clave para atraer la atención de los buscadores. Conviene no detectar páginas que contienen este tipo de términos como relevantes, puesto que en realidad no lo son.

Hay algoritmos diseñados específicamente con el propósito de tener un buen resultado frente a las situaciones anteriores, como el algoritmo de Page Rank de Google, que asigna un valor numérico, un peso, a cada enlace con el objetivo de medir la importancia de ese sitio con respecto al resto de sitios analizados.

Con la aparición de la web 2.0, el análisis de la estructura de los sitios web se ha hecho más complejo, al existir páginas con mucho más contenido dinámico y dependiente de las acciones de los usuarios. Esto hace que el esquema de relación entre los diferentes enlaces no pueda ser estático, sino que cambie con el paso del tiempo.

En este sentido, existen estudios[5] que afirman que la noción de “página” dentro de un sitio web está perdiendo importancia debido a la forma de construcción de las webs. Hoy en día es posible explorar todo el contenido de un sitio a través de una sola página sobre la que se aplican diferentes parámetros a través de la URL para cargar dinámicamente el contenido. Es por ello que el análisis de la estructura de páginas ha de ser en este aspecto tenido en cuenta y no considerar que el sitio web esté formado por una sola página.

[5] Extraído del informe *Dinamics of the Chilean web structure. Baeza-Yates.*

2.3.3 Contenido web

La Minería Web centrada en el contenido trata de analizar los datos que contiene la web. Estos datos pueden estar representados de muchas formas distintas y provenir de diversas fuentes. Además, estos datos pueden cambiar de forma dinámica, como se ha mencionado en apartados anteriores, por lo que parte del contenido de la página puede ser distinto en función de una consulta de datos realizada.

Todos estos datos no suelen estar ordenados, o la estructura depende de la página que se consulte, y como se ha comentado es posible que estén representados de formas muy diversas: mediante texto, imágenes, audio, video u otros enlaces, por ejemplo, e incluso puede haber contenido que directamente esté oculto para los usuarios en general, como los metadatos.

En la siguiente tabla se muestra un resumen de los diferentes tipos de contenidos disponibles, así como su descripción y el tipo de fuente:

Tipo de Contenido	Descripción	Fuente
Artículos	Texto relacionado con un tema particular. Acostumbra a tener título, cuerpo y en ocasiones subtítulo.	Generado a partir de un profesional, una agencia de noticias, un particular o agregado de otros sitios.
Productos	Items vendidos en un sitio. Típicamente tienen título, descripción, palabras clave, puntuación y otros atributos como el precio, el fabricante o la disponibilidad en función de la región.	Generado desde la propia administración del sitio o enlazado a partir de otros sitios web como *Ebay* o *Amazon*.
Blogs	Espacios web en los que una persona escribe acerca de un tema en particular y otros usuarios pueden opinar añadiendo comentarios y enlazar el sitio.	Administración del sitio, empleados de una empresa o generado de forma automática.
Wikis	Herramienta de colaboración *online* en la que los usuarios pueden añadir o editar contenidos de forma rápida y fácil.	Generalmente administrada por los propios usuarios.
Foros	Espacios web en los que se plantean preguntas y otros usuarios pueden contestar a las mismas. Pueden añadir elementos de valoración de las respuestas, así como otros datos relativos a las preguntas: categoría, fecha de creación, etc.	Las respuestas de los foros suelen ser contestadas por los mismos usuarios. También los hay de expertos en una materia, en los que son estos profesionales los que administran el contenido del foro y contestan las preguntas.

Fotografías y Vídeos	Elementos multimedia en forma de videos y fotos. Pueden ser tanto de carácter profesional como doméstico.	Generadas tanto por usuarios domésticos como por profesionales, según el tipo.
Encuestas	Encuestas generadas acerca de temas específicos con un conjunto de posibles respuestas limitadas.	Preguntas planteadas por profesionales del sitio y contestadas por los usuarios.
Páginas de Perfil	Páginas de perfil de los usuarios. Normalmente generadas por los mismos usuarios en base a un conjunto de opciones limitado.	Generado por los usuarios de todo tipo.
Chats	Registros de conversaciones mantenidas por usuarios.	Conversaciones mantenidas por usuarios, ya sean expertos o no.
Revisiones	Revisiones sobre elementos o contenidos, que pueden ser cualquiera de los otros tipos de contenidos. Típicamente son productos.	Generados por profesionales o por usuarios no expertos.
Clasificados	Anuncios con una estructura determinada formados por un título y un cuerpo.	Generados a nivel profesional o por usuarios no expertos.
Listas	Listas de cualquier tipo de contenido combinado.	Generados por administradores o por usuarios no expertos.

Tabla 1: Tipos de Contenido - Fuente: Manning – Collective Intelligence in Action

De los anteriores, el análisis del contenido en forma de texto ha centrado el interés tanto de empresas como de universidades en los últimos años por ser el elemento que permite un análisis más “sencillo” o en mayor detalle. A pesar de ello el análisis de este elemento es muy complejo, ya que el texto de una página web no está estructurado por mucho que siga unas reglas básicas de jerarquía (al ser desarrollado mediante lenguaje HTML o XHTML).

Hay muchas formas de buscar información a través de Internet, pero si se piensa en “búsqueda” la primera idea es un buscador como Google, que aporta un listado de enlaces resultantes en base a una consulta en forma de texto. El listado de páginas resultantes tiene en su contenido todo o parte de la consulta realizada.

Tradicionalmente este tipo de buscadores tienen un conjunto de herramientas[6] que recorren las diferentes webs en busca de contenido o palabras claves que puedan ser útiles en un proceso de búsqueda para almacenarlas e indexarlas, y proporcionar a los usuarios información precisa sobre sus consultas.

[6] Ver el apartado *3.*

Information Retrieval en este mismo trabajo.

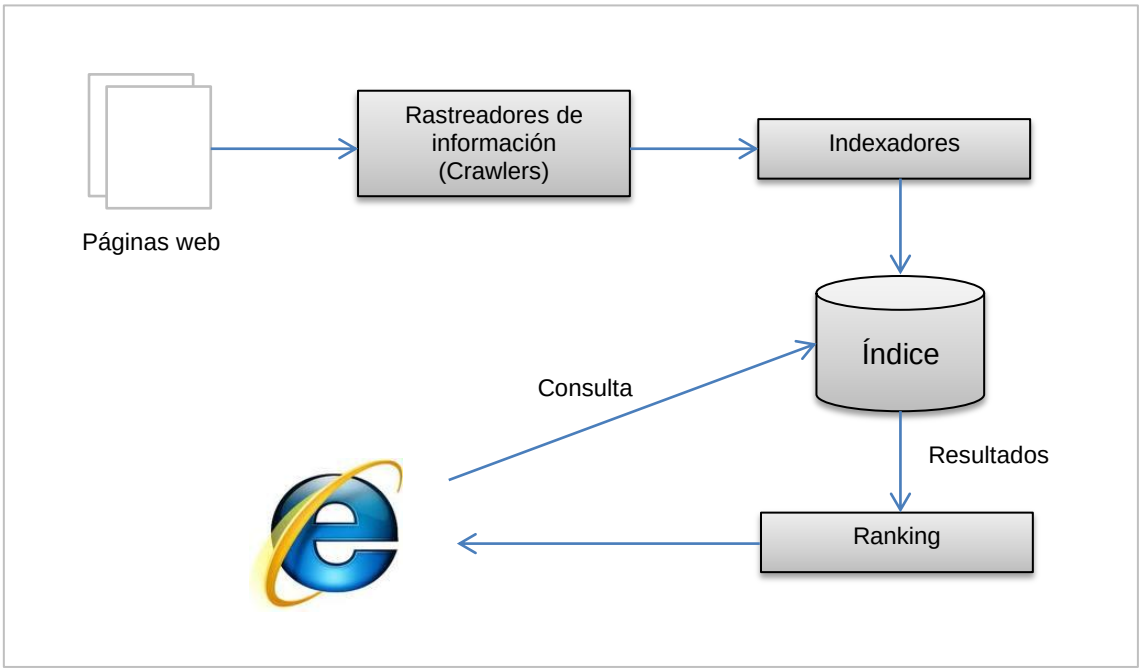


Ilustración 4: Proceso de Minería de Uso web. Fuente: Building an Intelligent web (Rajendra Akerkar, 2008)

Tal y como se muestra en el diagrama anterior, los buscadores realizan una búsqueda intensiva del contenido de Internet para almacenar un índice de los contenidos. Este índice se elabora teniendo en cuenta no sólo los contenidos de la web, sino las relaciones de las diferentes fuentes de datos entre ellas y las visitas contabilizadas a las páginas, entro otros parámetros.

Finalmente, la consulta de un usuario se optimiza internamente para encontrar las páginas indexadas que pueden ser más relevantes en el proceso de búsqueda, que formen el conjunto de enlaces de páginas resultante.

3 Information Retrieval

3.1 Introducción a Information Retrieval

La recuperación de información (del inglés Information Retrieval[7]) tiene como objetivo encontrar información relevante analizando contenidos de documentos, en nuestro caso páginas web.

Se trata de un conjunto de técnicas basadas no sólo en la recuperación de datos, sino en la extracción de la información y en el procesado de las gramáticas y de aspectos relativos al lenguaje natural para la extracción de conocimiento. Las técnicas de IR buscan entre los datos de entrada aquellos elementos que pueden ser de interés para el usuario, los etiquetan y les dan forma para ser presentados de un modo diferente al de la entrada.

Durante cientos de años la humanidad se ha dedicado a acumular conocimiento. Desde la Biblioteca de Alejandría hasta nuestros días las técnicas de almacenaje de la información se han perfeccionado, pero no han evolucionado a la misma velocidad las técnicas de búsqueda de esta información.

IR se basa principalmente en la búsqueda de información en forma de texto, aunque se pueden aplicar este tipo de mecanismos de extracción de datos sobre otros elementos. Por otro lado, no se puede aplicar una misma técnica para todo tipo de recursos a analizar, y el orden en que éstos se procesen no puede ser arbitrario.

En primer lugar, los datos de entrada han de ser procesados previamente, eliminando aquellos elementos que no aporten valor en el proceso de búsqueda, para luego analizar y clasificar los términos (que en el caso de procesamiento de texto serían palabras) que los componen en función del lenguaje a analizar.

Llegados a este punto, los datos obtenidos han de ser guardados de una forma determinada que facilite la obtención de conocimiento nuevo en fases posteriores del proceso de análisis, tanto en forma de consulta sobre el conjunto de información como en forma de patrones de “comportamiento” de estos datos.

Por último, la información extraída y almacenada ha de ser analizada, y este análisis ha de dar como resultado una evaluación del conocimiento extraído que permita determinar si las técnicas aplicadas durante todo el proceso han tenido los resultados esperados. En función de las conclusiones de este proceso hay que modificar el sistema de recuperación de la información para volver a iniciar el ciclo o adaptar las partes al tipo de información o fuente de datos de origen.

Como resultado, se obtiene un proceso de extracción y evaluación de la información que puede ser utilizado en múltiples campos. Entre las posibles alternativas de aplicación de este tipo de técnicas en el entorno web se pueden destacar:

- **Búsqueda de información**: La extracción de información de un conjunto de datos almacenados (ya sea en las mismas páginas, en bases de datos o en colecciones de documentos) es fundamental en el funcionamiento de un motor de búsqueda.

[7] En adelante, IR

- **Personalización de páginas**: Analizando los contenidos, el tipo de datos y la interacción de los usuarios con ellos se puede personalizar el contenido de las páginas para adaptarlas a las necesidades de los consumidores.

- **Análisis del comportamiento de usuarios**: Analizando otro tipo de datos, no únicamente el contenido, se pueden realizar cambios en la estructura web para adaptarla de un modo más natural al comportamiento de los usuarios mejorando la usabilidad de la web.

Tal y como se ha mencionado en apartados anteriores, la información publicada en Internet está completamente desestructurada. Este aspecto, sumado al crecimiento incesante de los datos publicados en la red un día tras otro, hace de los sistemas de IR unos elementos tremendamente atractivos en el entorno tecnológico actual.

3.2 Elementos de Information Retrieval

El proceso de extracción de la información se clasifica en diferentes fases secuenciales que definen los diferentes elementos de que se componen este tipo de técnicas. En términos generales, este proceso se descompone en los siguientes pasos:

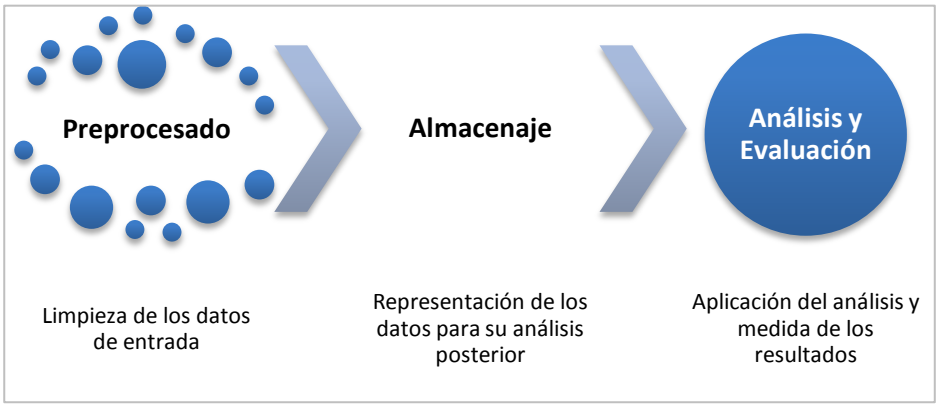


Ilustración 5: Proceso de extracción de la información.

- **Preprocesado**: Para obtener unos resultados adecuados a las necesidades del sistema los datos de la entrada casi nunca pueden ser analizados de forma directa. Por supuesto este aspecto depende del tipo de dato de entrada, pero por norma general es necesario un procesado previo de la información de entrada que permita una mejor interacción con las fases posteriores del análisis.

  Por lo tanto, dentro de este apartado se aplican un conjunto de técnicas de limpieza de los datos de entrada que tienen como objetivo eliminar aquellos elementos que no sean relevantes de cara al resultado final. Este proceso de eliminación de datos no ha de interferir en el valor obtenido durante el resto de fases del sistema, por lo que se han de evaluar aspectos como el dominio de datos de entrada o el tipo de análisis a aplicar, entre otros.

- **Almacenaje de la información**: La información extraída ha de ser almacenada. La representación que se haga de los datos, y por tanto el modo en que éstos sean almacenados, será crucial para determinar la relevancia de los datos obtenidos.

  De este modo, de nuevo teniendo en cuenta el tipo de datos y los resultados finales a obtener, habrá que aplicar una u otra técnica de representación que sirva para evaluar, por ejemplo, la importancia de un conjunto de palabras en un documento, la existencia o no de un conjunto de términos, o el orden de relevancia de los datos de entrada en función de la consulta hecha, entre otros.

Llegados a este punto, conviene añadir que el tipo de técnicas de Information retrieval que se explican en este proyecto se enfocan a la evaluación de unos datos en base a una consulta de entrada. Esta consulta, formada por un conjunto de términos introducidos por el usuario o como resultado de algún tipo de análisis previo, condiciona el procesamiento que se haga de los datos de entrada.

Por ello hay que tener en cuenta que la Representación de la Información ha de ir ligada en cierta forma con la consulta realizada, que también ha de ser representada. La conjunción de estas dos representaciones da como resultado una importancia relativa de todos los documentos o datos de la entrada con respecto a la consulta, y por lo tanto obtenemos una lista ordenada en función de la relevancia de cada documento.

Es decir, que se puede describir la relación de estos elementos como:

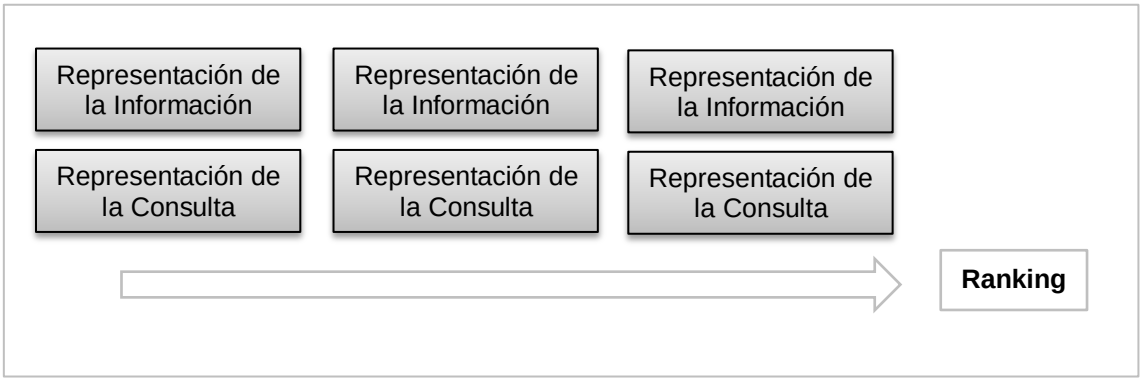


Ilustración 6: Proceso de representación de la Consulta.

El funcionamiento de las técnicas de recuperación de la información se basan en comparar el conjunto de palabras de la consulta realizada con el conjunto de palabras de cada uno de los documentos o fuentes de datos analizadas.

Para simplificar, si bien en el entorno web podríamos hablar de páginas como fuente de datos a analizar, en este proyecto se analizan las diferentes técnicas considerando las fuentes de datos como Documentos, independientemente de si el tipo de datos pertenece a webs o a otros entornos.

El modo en que se comparen las consultas con los documentos puede variar, por lo que en los siguientes apartados de explican los diferentes métodos de evaluación de consultas con respecto a documentos, así como la forma en que tanto las consultas como los documentos son almacenados.

Por otro lado, hay que considerar que el análisis produce unos resultados que varían tanto en función de la consulta como en función de los datos de los que se disponga. Una consulta más específica o compleja reducirá las coincidencias con el conjunto de documentos de entrada; del mismo modo, un conjunto de documentos de entrada tendrá por norma general una coincidencias menos favorables. La evaluación del tipo de representación de documentos y consultas tiene que tener este aspecto en cuenta. Los diferentes tipos de evaluación se explican también en apartados posteriores.

3.3 Técnicas de preprocesado

Tal y como se ha comentado anteriormente, la información no puede ser tratada de forma directa, sino que necesita de una procesamiento previo que eliminte aquellos elementos carentes de valor durante el proceso de recuperación de la información.

Este procesamiento requiere de una o varias técnicas que pueden ser aplicadas sobre los datos de entrada y de los que depende el resultado final del sistema.

3.3.1 Esquema de funcionamiento

El procesamiento previo está compuesto por diferentes técnicas que se aplican de forma secuencial sobre los datos de entrada para obtener un conjunto de términos que, manteniendo el contenido de los datos originales, tiene una forma más simple que favorece la aplicación de las técnicas de análisis posteriores.

La secuencia de procesamiento de los datos de entrada queda representada en el esquema siguiente:

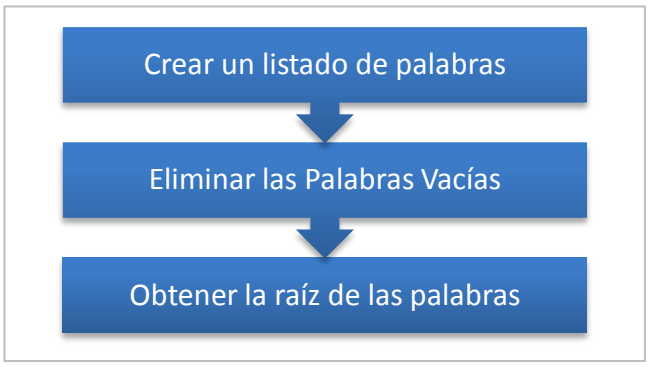


Ilustración 7: Secuencia de procesado de datos.

En primer lugar hay que leer el contenido en forma de texto de la entrada para obtener un conjunto de palabras. Por norma general, el texto de la entrada es un documento, que en el caso del entorno web contiene elementos propios del lenguaje HTML. Por ello, habrá que aplicar las búsquedas y sustituciones necesarias sobre el fichero de datos de entrada para eliminar todas las etiquetas de hipertexto que puedan dificultar el análisis posterior.

Una vez preprocesado el documento (sea o no del entorno web), se obtiene un documento formado por párrafos, formados a su vez por frases compuestas de palabras. De cada documento hay que procesar los fragmentos deseados (tal vez todo

el documento, tal vez sólo una parte dependiendo del tipo de análisis a realizar), para obtener finalmente un listado de palabras.

Este listado de palabras contiene información no relevante que hay que eliminar al no aportar valor (palabras vacías), para obtener finalmente un listado de palabras relevantes, borrando también cualquier carácter que pudiese interferir en el proceso de análisis (espacios, comas, saltos de línea o caracteres especiales).

Dicho listado es de nuevo procesado para obtener la forma invariable de cada palabra, ajena a cualquier tipo de derivación o tiempo verbal que pueda dificultar el proceso de búsqueda. El conjunto resultante de este último procesado inicial de los datos de entrada es la entrada para el resto de fases del sistema.

Este esquema puede aplicarse independientemente de la tipología de documentos de entrada y de su idioma. Sin embargo, dependiendo de éste último el funcionamiento de los algoritmos para eliminar las palabras vacías o para obtener la raíz de las palabras varía considerablemente. El dominio de datos del documento (esto es, la información de la que habla el texto) puede hacer variar también la forma en la que estas herramientas trabajan, simplificando el resultado final obtenido para concentrarse en el tipo de elementos a encontrar.

Por otro lado hay que tener en cuenta que, si bien éste esquema de funcionamiento es el más utilizado por los sistemas de recuperación de la información en el procesado previo de los datos, algunos buscadores web (soluciones comerciales que hacen uso de este tipo de técnicas) se saltan alguno de estos pasos previos, posiblemente para tener unos resultados más específicos.

3.3.2 Stop Words

Las palabras vacías, en ingles *Stop Words*, son aquellos términos contenidos en el texto que no aportan valor al conjunto de datos y que por lo tanto pueden ser eliminados de cara al proceso de análisis.

Por norma general, aquellas palabras que aparecen en un 80% de los documentos no sirven para distinguir unos documentos de otros y por lo tanto no aportan valor. Estos términos son artículos, preposiciones o conjunciones que hay que eliminar para tener como resultado un conjunto de “palabras clave” con el que trabajar.

Un ejemplo de listado de palabras vacías en español podría ser el siguiente:

de	su	son	estado	ellos	quienes	tú	tuyas	está
la	al	entre	desde	e	nada	te	suyo	estamos
que	es	está	todo	esto	muchos	ti	suya	estáis
el	lo	cuando	nos	mí	cual	tu	suyos	están
en	como	muy	durante	antes	sea	tus	suyas	esté
y	más	sin	estados	algunos	poco	ellas	nuestro	estés
a	pero	sobre	todos	qué	ella	nosotras	nuestra	estemos
los	sus	ser	uno	unos	estar	vosotros	nuestros	estéis
del	le	tiene	les	yo	haber	vosotras	nuestras	estén
se	ya	también	ni	otro	estas	os	vuestro	estaré
las	o	me	contra	otras	estaba	mío	vuestra	estarás

por	fue	hasta	otros	otra	estabamos	mía	vuestros	estará
un	este	hay	fueron	él	algunas	míos	vuestras	estaremos
para	ha	donde	ese	tanto	algo	mías	esos	estaréis
con	sí	han	eso	esa	nosotros	tuyo	esas	estarán
no	porque	quien	había	estos	mi	tuya	estoy	estaría
una	esta	están	ante	mucho	mis	tuyos	estás	…

Tabla 2: Listado de Palabras Vacías.

Estos listados dependen del lenguaje de los documentos de entrada, por lo que en caso de querer analizar datos de varios idiomas habría que detectar primero de qué lenguaje se trata para poder procesarlo con el fichero de palabras vacías adecuado según el caso. En algún caso podría depender también del dominio de datos, en el que la eliminación de una palabra que en otros entornos es trivial podría empeorar el resultado final.

Es por ello que, si bien se pueden encontrar listados de palabras vacías disponibles de forma pública, conviene revisarlos y ajustarlos a las necesidades del sistema a desarrollar, así como al tipo de procesamiento posterior que se va a realizar sobre los datos.

El funcionamiento de esta técnica es muy simple: se procesa palabra a palabra, y se coteja con el listado de palabras vacías. En el caso de encontrar la palabra en el listado de *Stop Words* el término es eliminado. En el caso que no se encuentre, se mantiene.

Un ejemplo de funcionamiento podría ser el siguiente:

Texto Original:

Nadie sabía que tuviese mujer ni hijos -cosa que puede suceder a la persona más decente del mundo-, ni parientes ni amigos -lo cual era en verdad algo más extraño-. Phileas Fogg vivía solo en su casa de Saville-Row, donde nadie penetraba. Un criado único le bastaba para su servicio. Almorzando y comiendo en el club a horas cronométricamente determinadas, en el mismo comedor, en la misma mesa, sin tratarse nunca con sus colegas, sin convidar jamás a ningún extraño, sólo volvía a su casa para acostarse a la media noche exacta, sin hacer uso en ninguna ocasión de los cómodos dormitorios que el Reform-Club pone a disposición de los miembros del círculo. De las veinticuatro horas del día, pasaba diez en su casa, que dedicaba al sueño o al tocador.

Fragmento de "La vuelta al Mundo en 80 días", de Julio Verne

Texto procesado sin Palabras Vacías:

Nadie sabía mujer hijos -cosa puede suceder persona más decente mundo-, parientes amigos -lo verdad más extraño-. Phileas Fogg vivía solo casa Saville-Row, nadie penetraba. criado único bastaba servicio. Almorzando comiendo club horas cronométricamente determinadas, mismo comedor, misma mesa, tratarse nunca colegas, convidar jamás ningún extraño, sólo volvía casa acostarse media noche exacta, hacer uso ninguna ocasión cómodos dormitorios Reform-Club pone disposición miembros círculo. veinticuatro horas día, pasaba diez casa, dedicaba sueño tocador.

Fragmento de "La vuelta al Mundo en 80 días", de Julio Verne – Sin Palabras Vacías

El procesado de las Palabras Vacías del ejemplo anterior se ha hecho haciendo uso de un fichero público de Stop Words, del que se ha mostrado un fragmento anteriormente. La ejecución del algoritmo de eliminación de palabras vacías se ha hecho mediante código PHP en un servidor local.

Como se puede observar, se ha reducido la longitud y complejidad del texto eliminando palabras vacías del texto original teniendo en cuenta un listado de palabras vacías. El resultado es un texto un poco más difícil de entender a la lectura pero que carece de términos que no aporten valor al análisis final.

3.3.3 Stemming

El proceso de Stemming es un análisis palabra a palabra del texto de entrada para eliminar aquellas partes que pueden resultar confusas o que acotan demasiado el término a buscar.

Consiste en obtener la raíz de las palabras analizando la estructura de cada una de ellas, teniendo en cuenta aspectos específicos del lenguaje con el que están escritos para eliminar o cambiar aquellas partes que limitan su significado.

El ejemplo siguiente ilustra el funcionamiento de este proceso:

Palabras originals: Documentadas lluvias torrenciales

Palabra tras Stemming: Documentada lluvia torrencial

Se eliminan todas aquellas partes de la palabra que denoten tiempo verbal, plurales o gerundios, por ejemplo, para conseguir la raíz del término dado.

Como se puede ver este proceso es totalmente dependiente de los aspectos gramaticales del lenguaje y ha de hacerse en varios pasos. El resultado final no es el mejor en todos los casos, pero el porcentaje de fallos es relativamente bajo (depende del algoritmo utilizado y de su implementación) y en caso de excepciones no contempladas, éstas se pueden tratar aparte.

Hay varios algoritmos de Stemming desarrollados y con un resultado similar, pero el más popular y por tanto el que se explica a continuación es el desarrollado por Martin Porter en 1980, conocido como aloritmo de Porter[8]. Este algoritmo, desarrollado originalmente para el idioma Ingles, y teniendo en cuenta por tanto sus particularidades gramaticales, ha sido adaptado a la gramática española haciendo uso del Framework Snowball[9]. Ésta última adaptación es la que se explica a continuación, referenciada en la página oficial del autor.

El algoritmo de Porter divide el proceso de extracción de la raíz en una serie de pasos diseñados para eliminar los prefijos y los sufijos de forma ordenada, además de hacer las transformaciones necesarias en la palabra para obtener la raíz.

En primer lugar hay que definir lo que el algoritmo de Stemmer en español define como vocales las letras *a, e, i, o, u,* así como sus formas acentuadas *á, é, í, ó, ú* y la diéresis *ü*. Al analizar una palabra se buscan las vocales y consonantes que la componen, buscando las combinaciones de cada una de ellas y realizando las sustituciones en consecuencia.

[8] Información disponible a través de la página oficial del algoritmo, propiedad del propio Martin Porter: http://tartarus.org/~martin/PorterStemmer/

[9] Se trata de un lenguaje de programación para la gestión de Strings que es muy utilizado en el diseño de algoritmos Stemming a través de Scripts con una sintaxis específica. El código desarrollado en Snowball se puede compilar para ser traducido a lenguajes como Java o C.

El ejemplo siguiente ilustra las combinaciones de Vocales y Consonantes con las que se juega para realizar las transformaciones:

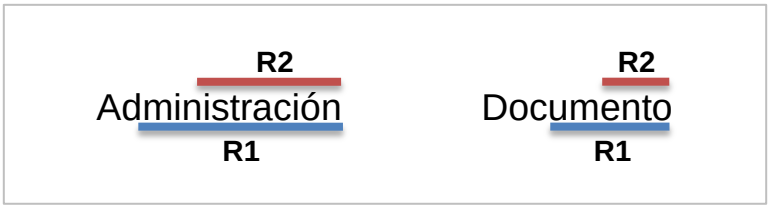
Palabras	Combinaciones
Pancarta	CVCCVCCV
Trabajo	CCVCVCV

Las combinaciones sobre las que se transforma la palabra suelen depender de la secuencia VC (vocal y consonante) en la palabra, por lo que este patrón es el más buscado a partir del cual se definen subgrupos dentro de la palabra.

Los posibles subgrupos dentro de cada palabra se distribuyen en las 3 estructuras siguientes, teniendo en cuenta que las palabras empiezan siempre por la parte izquierda, siguiente el orden de lectura.

- **R1**: Es la parte de la palabra (conjunto de letras) que hay después de la primera consonante situada después de una vocal (combinación VC). Si no se encuentra esta consonante entonces es un conjunto vacío.

- **R2**: Es la parte de la palabra (conjunto de letras) que hay después de la primera consonante dentro de R1 (VC dentro de R1). Si no existe esta combinación entonces es conjunto vacío.



- **RV**: Si la segunda letra es una consonante, entonces es la parte de la palabra (conjunto de letras) después de la siguiente vocal. Si las dos primeras letras de la palabra son vocales, entonces RV es la región después de la siguiente consonante, y si no (la palabra empieza por CV) entonces es la región después de la tercera letra. Si ninguna de las anteriores condiciones se cumple, RV es conjunto vacío.

Así pues, tomando como referencia los conjuntos anteriores el algoritmo se define siguiendo una serie de pasos secuenciales, en los que se ejecutan las acciones que a continuación se muestran:

Paso 0: Sustitución de pronombres.

- Eliminar la lista de pronombres **me**, **se**, **sela**, **selo**, **selas**, **selos**, **la**, **le**, **lo**, **las**, **les**, **los**, **nos** y eliminarlos si están dentro de RV y si éstos vienen precedidos por alguna de las combinaciones siguientes:

    - **iéndo**, **ándo**, **ár**, **ér**, **ír**
    - **ando**, **iendo**, **ar**, **er**, **ir**
    - **yendo** después de una **u**

  En el primero de los casos hay que eliminar también el acento.

  Ejemplo:

Cant**ándosela** → Cant**ando**

Paso 1: Eliminación de sufijos.

- Buscar los sufijos **anza**, **anzas**, **ico**, **ica**, **icos**, **icas**, **ismo**, **ismos**, **able**, **ables**, **ible**, **ibles**, **ista**, **istas**, **oso**, **osa**, **osos**, **osas**, **amiento**, **amientos**, **imiento**, **imientos** y eliminarlos si están dentro de la región R2.

  Ejemplo:

Montañ**osos** → Montañ

- Buscar los sufijos **adora**, **ador**, **ación**, **adoras**, **adores**, **aciones**, **ante**, **antes**, **ancia**, **ancias** y eliminarlos si están dentro de la región R2. Si están precedidos por **ic** borrarlos también si están dentro de R2.

  Ejemplo:

Pronost**icadores** → Pronost

- Buscar los sufijos **logía**, **logías** y reemplazarlos por **log** si están en R2.

  Ejemplo:

Traumato**logía** → Traumato**log**

- Buscar los sufijos **ución**, **uciones** y reemplazarlos por **u** si están en R2.

  Ejemplo:

Dismin**ución** → Disminu

- Buscar los sufijos **encia**, **encías** y reemplazarlos por **ente** si están en R2.

  Ejemplo:

Equival**encia** → Equival**ente**

- Buscar los sufijos **amente** y borrarlos si están en R1. Se están precedidos por **iv** entonces borrarlos si están en R2 (de ser así, si están precedidos por **at** borrarlos si están en R2), y si no si están precedidos por **os**, **ic** o **ad**, borrarlos si están en R2.

  Ejemplo:

Carit**ativamente** → Carit

- Buscar los sufijos **mente** y borrarlos si están en R2. Si además están precedidos por **ante**, **able** o **ible**, borrarlos si están en R2.

  Ejemplo:

Inconsol**ablemente** → Inconsol

- Buscar los sufijos **idad**, **idades** y borrarlos si están en R2. Si están precedidos por **abil**, **ic** o **iv** borrarlos si están en R2.

  Ejemplo:

Irrespons**abilidad** → Irrespons

- Buscar los sufijos **iva**, **ivo**, **ivas**, **ivos** y borrarlos si están en R2. Si además vienen precedidos por **at**, borrarlos si están en R2.

  Ejemplo:

Superlativo → Superl

Paso 2: Eliminación de sufijos en verbos.

Este paso se divide en dos partes, parte A y parte B. En caso de que el primer paso no hay logrado eliminar ninguna parte de la palabra, entonces se pasa al apartado 2A. Si se ha entrado en el 2A pero no se ha eliminado nada, entonces se pasa al paso 2B.

- **2A**: Eliminar todos los sufijos de los verbos que empiecen (los sufijos) por **y**. Es decir, buscar los siguientes sufijos en RV y borrarlos si vienen precedidos de **u**: **ya**, **ye**, **yan**, **yen**, **yeron**, **yendo**, **yo**, **yó**, **yas**, **yes**, **yais**, **yamos**. La combinación de la **u** y alguno de los sufijos anteriores no ha de estar juntamente en RV, tan sólo el sufijo.

  Ejemplo:

Reconstr**uyendo** → Reconstru

- **2B**: Ejecutar las acciones siguientes:

  - Buscar los sufijos **en**, **es**, **éis**, **emos** y borrarlos, y si están precedidos por **gu** borrar la **u**.

    Ejemplo:

Cant**en** → Cant

  - Buscar los sufijos **arían**, **arías**, **arán**, **arás**, **aríais**, **aría**, **aréis**, **aríamos**, **aremos**, **ará**, **aré**, **erían**, **erías**, **erán**, **erás**, **eríais**, **ería**, **eréis**, **eríamos**, **eremos**, **erá**, **eré**, **irían**, **irías**, **irán**, **irás**, **iríais**, **iría**, **iréis**, **iríamos**, **iremos**, **irá**, **iré**, **aba**, **ada**, **ida**, **ía**, **ara**, **iera**, **ad**, **ed**, **id**, **ase**, **iese**, **aste**, **iste**, **an**, **aban**, **ían**, **aran**, **ieran**, **asen**, **iesen**, **aron**, **ieron**, **ado**, **ido**, **ando**, **iendo**, **ió**, **ar**, **er**, **ir**, **as**, **abas**, **adas**, **idas**, **ías**, **aras**, **ieras**, **ases**, **ieses**, **ís**, **áis**, **abais**, **íais**, **arais**, **ierais**, **aseis**, **ieseis**, **asteis**, **isteis**, **ados**, **idos**, **amos**, **ábamos**, **íamos**, **imos**, **áramos**, **iéramos**, **iésemos**, **ásemos** y borrarlos.

Ejemplo:

Bail**aron** → Bail

Paso 3: Otros sufijos.

Finalmente, el paso 3 consiste en la eliminación de otros sufijos siguiendo las acciones siguientes:

- Buscar los sufijos **os**, **a**, **o**, **á**, **í**, **ó** y borrarlos si están en RV.

  Ejemplo:

Cant**ó** → Cant

- Buscar los prefijos **e** y **é** y borrarlos si están en RV. Si además vienen precedidos por **gu** estando la **u** en RV, borrar la **u**.

  Ejemplo:

Cant**ó** → Cant

- Como última acción, se eliminan los acentos.

3.4 Métodos de representación

3.4.1 Term-Document Frequency Matrix

Una Term-Document Frequency Matrix en una forma de representar una colección de documentos mediante una matriz.

Se entiende como *colección de documentos* al conjunto de elementos de texto que se quiere tratar, dividiéndolo en partes que pueden ser frases (como en el caso de este proyecto) o documentos enteros en el caso de otro tipo de análisis.

Cada uno de estos documentos (o elementos de texto a tratar) está compuesto por un conjunto de términos a analizar y que forman el conjunto de columnas de la matriz a representar. Por otro lado, que cada uno de los documentos a analizar es una fila. Los conjuntos de filas y de columnas se tratan como arrays empezando en la posición 0.

Por tanto, el conjunto de documentos estaría formado, por ejemplo, por D0, D1, D2, D3, etc. mientras que el conjunto de palabras lo formaría el conjunto P1, P2, P3, etc.

El valor de las celdas de la matriz contiene el número de veces que cada palabra aparece en cada uno de los documentos. Un ejemplo de esta matriz queda ilustrado en la tabla 3.

Suele ocurrir en este tipo de representación que el número de palabras (y por lo tanto el número de columnas) es mucho más elevado que el número de filas, por lo que la matriz resultante es dispersa. En este caso se suele almacenar valor únicamente en aquellas celdas donde el número de palabras es mayor que 0.

	P0	P1	P2	P3	P4	P5
D0	0	0	3	2	5	1
D1	2	5	0	0	2	0
D2	3	0	4	2	1	0
D3	0	1	3	2	0	1
D4	2	1	0	1	0	2

Tabla 3: Term-Document Frequency Matrix.

De aquí se deriva también un tipo de representación de esta matriz, que consiste en usar tripletas compuestas por la fila, la columna y el valor que contiene:

(número de fila, número de columna, número de apariciones)

De este modo, la matriz anterior podría representarse como:

(0,2,3) (2,0,3) (3,5,1)
(0,3,2) (2,2,4) (4,0,2)
(0,4,5) (2,3,2) (4,1,1)
(0,5,1) (2,4,1) (4,3,1)
(1,0,2) (3,1,1) (4,5,2)
(1,1,5) (3,2,3)
(1,4,2) (3,3,2)

Al ampliar estos datos a un ejemplo real y obtener una medida verídica del número de palabras en un documento y el número de coincidencias a través de una matriz, esta segunda representación de los datos muestra claramente cómo se reduce el espacio necesario para almacenar la información de aparición de las palabras en cada documento.

Por ejemplo, si se realiza el análisis de 200 palabras contenidas en documentos, se necesitarías 200 palabras x 3 = 600 posiciones. De otro modo, si de esta matriz se consideran únicamente las coincidencias entre palabras y documentos, representando la información mediante tripletas, el número de datos a almacenar es mucho menor.

3.4.2 Term-Document Weight Matrix

Existe otra forma, similar a la explicada en el apartado anterior, para almacenar la existencia de las palabras en documentos. Se trata de nuevo de utilizar una matriz, pero en este caso el valor no es el número de veces que las palabras aparecen en los documentos, sino la importancia relativa que tiene cada palabra en cada documento, con respecto al máximo de veces que aparece la palabra más utilizada.

Es decir, se aplica la siguiente fórmula para obtener el valor de cada celda:

$$W_{ij} = \frac{Frecuencia_{ij}}{\underset{k=1}{\overset{m}{MAX}} Frecuencia_{ik}}$$

Ecuación 1: Peso de los términos.

W_{ij} representa el peso en el documento de la palabra *ij*, donde de la fila *i* identifica el documento y *j* representa la palabra de la que se busca el peso. La frecuencia de *ij* representa las veces que la palabra *j* aparece en el documento *i*. Así pues, teniendo en cuenta que *m* representa el máximo número de palabras, la MAX Frecuencia *ik* corresponde a la frecuencia máxima de todas las palabras del documento *i*.

De este modo, la matriz del ejemplo anterior (*Tabla 1: Term-Document Frequency Matrix*) quedaría representada de la siguiente forma:

	P0	P1	P2	P3	P4	P5
D0	0.00	0.00	0.60	0.40	1.00	0.20
D1	0.40	1.00	0.00	0.00	0.40	0.00
D2	0.75	0.00	1.00	0.50	0.25	0.00
D3	0.00	0.33	1.00	0.67	0.00	0.33
D4	1.00	0.50	0.00	0.50	0.00	1.00

Tabla 4: Term-Document Weight Matrix

Este tipo de matriz puede resultar de gran utilidad para entender la importancia de cada palabra dentro del documento, y poder descubrir por ejemplo la temática o los términos más importantes de cada uno de ellos.

3.5 Métodos de análisis y evaluación

3.5.1 Boolean Retrieval Model

Tomando como referencia la representación de documentos basada en una *Term-Document Matrix* vista en apartados anteriores, este modelo es de gran utilidad para realizar consultas sobre documentos.

En lugar de guardar el número de veces que cada palabra aparece en un documento, o el peso que representa con respecto al total de palabras que aparece, *Boolean Retrieval Model* se basa únicamente en el hecho de si una palabra aparece (valor de 1) o no aparece (valor de 0) en cada documento.

Estos valores, a modo de cierto o falso, se almacenan del mismo modo en una matriz, que contienen tantas filas como documentos y tantas columnas como palabras a buscar. Siguiendo con el ejemplo de apartados anteriores[10], se puede representar la colección de documentos de la siguiente forma:

	P0	P1	P2	P3	P4	P5
D0	0	0	1	1	1	1
D1	1	1	0	0	1	0
D2	1	0	1	1	1	0
D3	0	1	1	1	0	1
D4	1	1	0	1	0	1

Tabla 5: Boolean Term-Document Matrix.

[10] Ver *Tabla 3: Term-Document Frequency Matrix*

Este modelo utiliza operaciones lógicas sobre boleanos (or, and y not) para realizar consultas de aparición de palabras en documentos. Así, por ejemplo, para saber si un documento contiene las palabras P2 y P5, la consulta a realizar sería *P2 and P5.*

Para realizar esta operación se tendrían que obtener los conjuntos de documentos en los que aparecen cada una de estas palabras. Es decir, el **CjtoP2**, formado por {D0,D1,D3} y el **CjtoP5** formado por {D0,D3,D4}.

De este modo, la operación lógica sería:

CjtoP2 *and* **CjtoP5** = {D0,D3}

Este tipo de modelo de consulta es muy útil por la posibilidad de utilizar cualquier tipo de combinación de operaciones lógicas, lo cual facilita la búsqueda por su simplicidad y eficiencia. Además, el hecho de usar valores binarios para la representación también supone un ahorro del espacio necesario para guardar la colección de documentos.

Sin embargo, también tiene desventajas, y es que, por ejemplo, no se puede distinguir la relevancia de las palabras en los documentos, y en ocasiones el número de resultados es muy elevado como para poderse utilizar sin algún procesado previo (P3 *or* P4 sería un ejemplo de este caso, donde se retornaría todo el conjunto de documentos).

3.5.2 Vector Space Model

Se trata de otro modelo de consulta, más preciso que el *Boolean Retrieval Model* al trabajar con una matriz de frecuencias (*Term-Docuemnt Frequency Matrix*). Este modelo representa las consultas y los documentos como vectores.

Representando el número de documentos a analizar como *N* y el número de palabras como *m*, entonces la matriz de pesos sería de dimensión *Nxm*.

Por otro lado, dada la siguiente representación del vector:

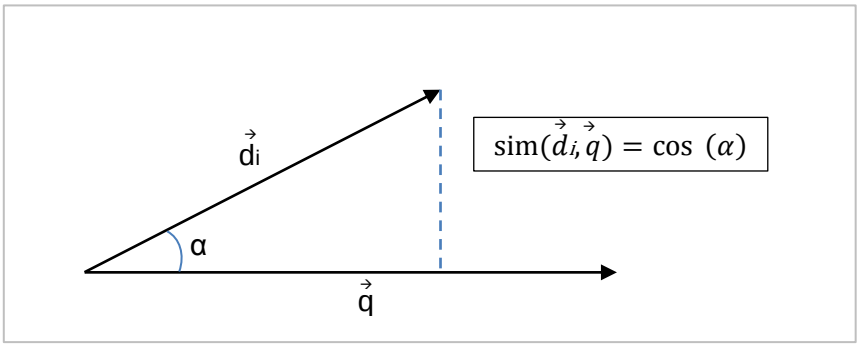


Ilustración 8: Representación gráfica de consultas y documentos.

Donde *d* representa cada uno de los documentos, representando en forma de vector a partir de la *Term-Document Weight Matrix*, usando la representación de:

$$d_i = (w_{i1}, w_{i2}, w_{i3}, w_{i4}, \ldots)$$

Donde w_{i1} representa el peso de la palabra P1 en el documento *i*. Por ejemplo, d_0 sería representado por el siguiente vector:

$$d_0 = (0,0,0.60,0.40,1.00,0.20)$$

Por otro lado, respecto a la representación gráfica, *q* representa mediante la misma representación la consulta realizada:

$$q = (w_{i1}, w_{i2}, w_{i3}, w_{i4}, \ldots)$$

Teniendo lo anterior en cuenta, se puede definir una función de similitud *sim(d$_i$,q)* que represente la distancia entre los vectores que representan el documento y la consulta. Esta medida es en realidad el *cos* del ángulo que forman estos vectores, que se puede obtener mediante el módulo y el producto de los mismos dos vectores:

$$sim(di, q) = \frac{\vec{di} \cdot \vec{q}}{|\vec{di}| \text{ x } |\vec{qi}|} = \frac{|\vec{di}| \text{ x } |\vec{qi}| \text{ x } \cos(\alpha)}{|\vec{di}| \text{ x } |\vec{qi}|} = \cos(\alpha)$$

Ecuación 2: Función de similitud.

Mediante este método las consulta se pueden relacionar con el peso de cada término buscado, y los documentos no se clasifican según si contienen el término o no, sino mediante un ranking de la similitud con respecto a la consulta.

3.5.3 Precision y Recall

Para evaluar el rendimiento del modelo de consulta se utilizan varias métricas que comparan el resultado obtenido con el resultado esperado. *Precision* y *Recall* forman parte de este tipo de métricas.

Suponiendo *R* el conjunto de documentos relevantes (es decir, aquellos que se querrían recuperar), y *A* el conjunto de documentos recuperado, se puede definir cada una de estas medidas como:

$$precision = \frac{|\, R \cap A \,|}{|\, A \,|}$$

$$recall = \frac{|\, R \cap A \,|}{|\, R \,|}$$

Ecuación 3: Precision y Recall.

La precisión es una medida del rigor de la consulta. Un valor de precisión alto significa que la probabilidad de que un documento recuperado sea relevante es también alta.

Por otro lado, *recall* indica qué porcentaje de documentos relevantes se han recuperado. Si el valor de esta métrica se acerca a uno, entonces es que se han recuperado la mayor parte de documentos relevantes de la colección de documentos analizada.

Normalmente estas dos métricas se complementan la una a la otra. Se puede incrementar el *recall* incrementando el número de documentos recuperado. Sin embargo, esta medida puede hacer aumentar el número de documentos no relevantes recuperados y por lo tanto reducir la *precision*.

4 Repositorios de Videos Online

4.1 Estudio de principales repositorios

Actualmente la oferta de repositorios de videos online es elevada y variada. Hay repositorios temáticos, centrados en categorías concretas, e incluso los hay que no sólo muestran contenido aportado por los usuarios sino que producen sus propios contenidos.

Con el objetivo de tener una visión global de los repositorios de videos más destacados a día hoy se ha sometido a algunos de estos portales a un breve cuestionario referente a las funcionalidades ofrecidas, la tipología de videos mostrados o los contenidos asociados a los videos.

Es importante tener en cuenta que el listado que a continuación se detalla se ha realizado pensando en tener un conjunto de repositorios con funcionalidades diversas. En caso de hacer un ranking de visitas de los más populares, algunos de estos reproductores no estarían presentes. Se han analizado, no obstante, para tener un abanico mayor de posibles funcionalidades ofrecidas.

Entre estos portales podemos destacar[11]:

Nombre del Portal	Enlace
3alacarta	http://www.3alacarta.cat/
AOL Video	http://video.aol.com/
Beet.tv	http://www.beet.tv/
Blinkx	http://www.blinkx.com/
blip.tv	http://www.blip.tv/
Break.com	http://www.break.com
Crackle	http://www.crackle.com
Dailymotion	http://www.dailymotion.com
ElTerrat.tv	http://www.elterrat.tv/
Fearnet	http://www.fearnet.com
Hulu	http://www.hulu.com
Joost	http://www.joost.com
Megavideo	http://www.megavideo.com/
Metacafe	http://www.metacafe.com/
OneWorldTV	http://tv.oneworld.net/
RTÉ player	http://www.rte.ie/player/
RuTube	http://rutube.ru/
Tudou	http://www.tudou.com/
Veoh	http://www.veoh.com/
Vimeo	http://vimeo.com/
Yahoo! Video	http://video.yahoo.com/
Youku	http://www.youku.com/
YouTube	http://www.youtube.com/

Tabla 6: Listado de repositorios.

[11] Listado ordenado alfabéticamente.

Mencionar que hay otros repositorios que, por razones de limitación geográfica (únicamente disponibles en países de origen) no han podido ser analizados, pero que conviene tener en cuenta en la medida de lo posible: *BBC iPlayer*, *BigPond TV* y *TV Catchup* son algunos ejemplos. El conjunto de atributos analizados y los resultados obtenidos quedan reflejados en la Tabla 7 que se muestra en la pròxima pàgina.

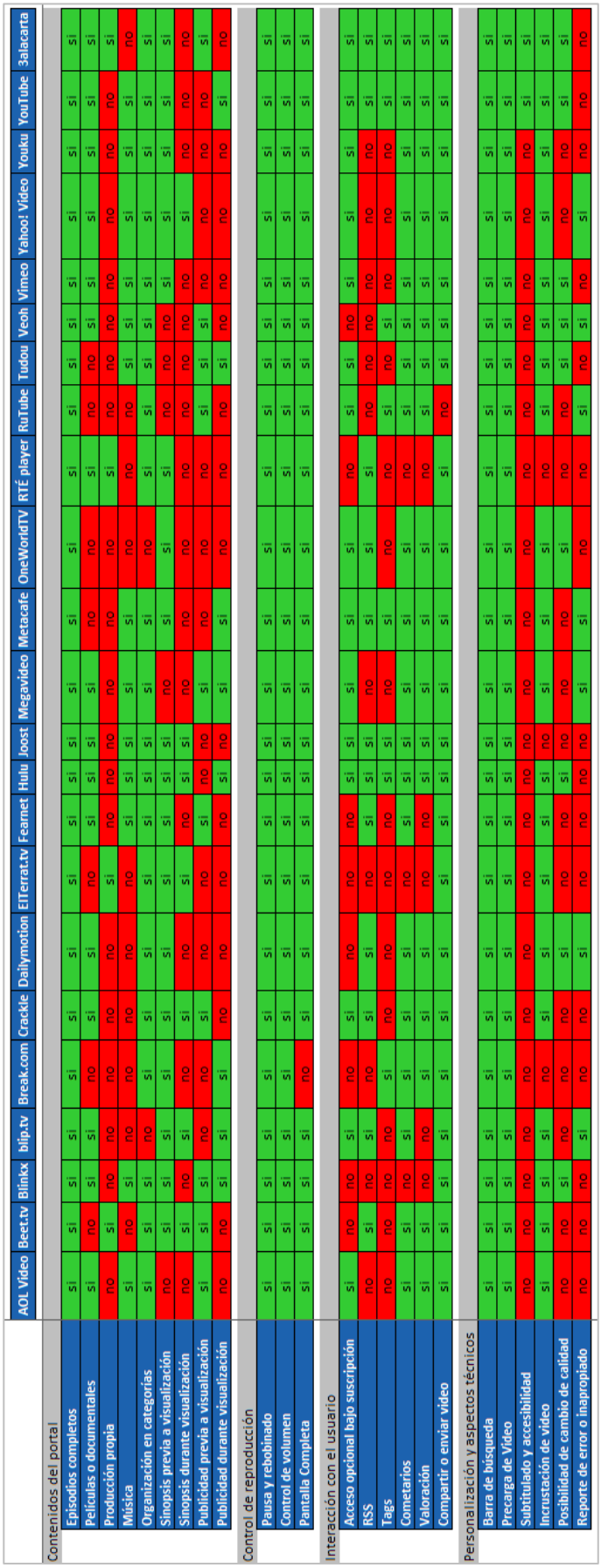
Cada apartado de la tabla ha sido considerado teniendo en cuenta los aspectos siguientes:

1. Preguntas relativas a contenidos del portal:

- **Episodios completos**: Hace referencia a si se presentan episodios de video completos en el sitio, o bien si se trata de fragmentos o vistas previas de los contenidos.
- **Películas o documentales**: Inclusión o no en el portal de películas o documentales de producción propia, entendiendo este contenido de carácter gratuito para el usuario.
- **Producción propia**: Indica si los contenidos son producidos o editados por la administración del portal, o bien si (por el contrario) son adquiridos a otras productoras o aportados por usuarios.
- **Música**: Oferta de música en el catálogo de contenidos del sitio web, dedicando un apartado específico del sitio (canal, sección, etc.) a este tipo de contenidos.
- **Organización en categorías**: Indica si el portal dispone o no de una organización del catálogo de videos en categorías que facilite la navegación del usuario.
- **Sinopsis previa a visualización**: Indica si en el momento previo a la visualización del video se puede disponer de un resumen del contenido a visualizar, ya sea en la propia área de visualización como en una región próxima a ella.
- **Sinopsis durante visualización**: Indica durante la visualización del video se puede disponer de un resumen del contenido a visualizar, ya sea en la propia área de visualización como en una región próxima a ella.
- **Publicidad previa a visualización**: Indica si en el momento previo a la reproducción se muestra un elemento publicitario que durante un período de tiempo (tiempo que en caso de ser publicidad audiovisual cubre toda la reproducción del spot o parte de ella) no puede ser evitado por parte del usuario. Este elemento publicitario puede estar situado en la propia área de reproducción o junto a ella.
- **Publicidad durante visualización**: Indica si durante la visualización se muestra un elemento publicitario en la propia área de reproducción, pueda ser evitado por parte del usuario o no.

2. Preguntas relativas al control de la reproducción:

- **Pausa y rebobinado**: Indica si el reproductor de videos contiene controles de pausa y la capacidad de avanzar el video hasta un punto concreto sin tener que esperar a la reproducción de todo el contenido hasta ese punto.
- **Control de volumen**: Indica si el reproductor contiene controles de control del volumen.
- **Pantalla Completa**: Indica si el reproductor contiene controles de visualización de video a pantalla completa.



		AOL Video	Beet.tv	Blinkx	blip.tv	Break.com	Crackle	Dailymotion	EiTerrat.tv	Fearnet	Hulu	Joost	Megavideo	Metacafe	OneWorldTV	RTÉ player	RuTube	Tudou	Veoh	Vimeo	Yahoo! Video	Youku	YouTube	3alacarta
Contenidos del portal	Episodios completos	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si
	Películas o documentales	si	no	si	si	no	si	si	no	si	si	si	si	no	no	si	no	no	si	si	si	si	si	si
	Producción propia	no	si	no	no	no	no	no	si	no	no	no	no	no	no	si	no	no	no	no	no	no	no	si
	Música	si	no	si	no	no	no	no	no	si	si	si	si	si	no	no	no	si	si	si	si	si	si	no
	Organización en categorías	si	si	si	no	si	si	si	si	si	si	si	si	si	no	si	si	si	si	si	si	si	si	si
	Sinopsis previa a visualización	no	si	si	si	si	si	si	si	si	si	si	no	si	si	si	no	no	no	si	si	si	si	si
	Sinopsis durante visualización	no	si	no	si	no	si	no	si	no	si	si	no	no	no	no	no	no	no	no	si	no	no	no
	Publicidad previa a visualización	si	si	si	no	no	si	no	no	si	no	no	si	no	no	no	si	si	si	no	no	no	no	si
	Publicidad durante visualización	no	no	si	si	si	no	no	no	no	si	no	si	si	no	no	no	si	no	no	no	no	si	no
Control de reproducción	Pausa y rebobinado	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si
	Control de volumen	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si
	Pantalla Completa	si	si	si	si	no	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si
Interacción con el usuario	Acceso opcional bajo suscripción	si	no	no	si	no	si	no	no	no	si	si	si	si	si	no	si	si	no	si	si	si	si	si
	RSS	no	si	no	si	no	si	si	no	si	si	si	no	si	si	si	no	no	no	no	no	no	si	si
	Tags	no	no	no	no	si	no	no	no	no	si	si	no	si	no	no	si	no	si	no	no	no	si	si
	Cometarios	si	si	no	si	si	si	si	no	si	si	si	si	si	si	no	si	si	si	si	si	si	si	si
	Valoración	si	si	no	no	si	si	si	no	no	si	si	si	si	si	no	si	si	si	si	si	si	si	si
	Compartir o enviar video	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	no	si	si	si	si	si	si	si
Personalización y aspectos técnicos	Barra de búsqueda	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si
	Precarga de Video	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si	si
	Subtitulado y accesibilidad	no	no	no	no	no	no	no	no	no	no	no	no	no	no	no	no	no	no	no	no	no	si	si
	Incrustación de video	si	si	si	si	no	si	si	si	si	si	no	si	si	si	no	si	si	si	si	si	si	si	si
	Posibilidad de cambio de calidad	no	no	si	no	no	si	si	no	no	si	no	no	no	si	no	no	no	si	si	no	no	si	si
	Reporte de error o inapropiado	no	no	no	si	no	no	si	no	no	no	no	si	si	no	no	si	no	si	no	si	no	no	no

Tabla 7: Comparativa de repositorios.

3. **Preguntas relativas a la interacción con el usuario:**

- **Acceso opcional bajo suscripción**: Indica si el portal dispone de un área de acceso restringido bajo nombre de usuario y contraseña a través de la cual los usuarios pueden visualizar contenido adicional, o participar activamente en los contenidos del sitio.
- **RSS**: Indica si el portal contiene elementos de suscripción de contenidos a través de archivos de sindicación *RSS* o *Atom*.
- **Tags**: Indica si el catálogo de videos es clasificado adicionalmente mediante palabras claves (*tags*), que identifican el contenido del video.
- **Cometarios**: Indica si los usuario pueden participar de los contenidos del portal aportando comentarios acerca de los videos.
- **Valoración**: Indica si los usuarios pueden valorar los videos reproducidos a través de un rango de valor (típicamente de 1 a 5, donde 1 es inferior y 5 superior).
- **Compartir o enviar video**: Indica si se pone a disposición del usuario los elementos necesarios para compartir el video a través de una red social o enviar el video a un amigo.

4. **Preguntas relativas a personalización y aspectos técnicos:**

- **Barra de búsqueda**: Indica si el portal dispone o no de elementos de búsqueda situados en una zona fija del sitio, siempre visible y con capacidad de búsqueda en todo el catálogo de videos.
- **Precarga de Video**: Indica si el portal dispone de elementos de precarga automática de video a través de *Buffer*.
- **Subtitulado y accesibilidad**: Indica si el portal dispone de elementos que faciliten la accesibilidad de los usuarios con discapacidades sensoriales mediante posibilidad de aumento del contraste, modificación del tamaño de botones, inserción de subtítulos en la reproducción de videos u otros elementos similares. En el caso de hacer uso indirecto de un reproductor no propio del portal (*YouTube*) se ha considerado si el contenido publicado en el portal a analizar disponía de subtítulos.
- **Incrustación de video**: Indica si se ofrecen al usuario los elementos de código necesarios para la incrustar el contenido audiovisual y el reproductor en una página o elemento web externo al portal.
- **Posibilidad de cambio de calidad**: Indica si se pone a disposición del usuario controles sobre la calidad del video a reproducir mediante los que poder incrementar o disminuir la calidad de la reproducción según convenga (y en consecuencia consumir más o menos ancho de banda respectivamente).
- **Reporte de error o inapropiado**: Indica si el portal dispone de elementos para poder recibir de forma rápida comunicados por parte de los usuarios relativos al funcionamiento defectuoso o erróneo de un video, o bien notificaciones referentes a contenidos inapropiados.

Por otro lado, se interesante el análisis más en profundidad de alguno de los portales anteriores, teniendo en cuenta la tipología del sitio: producción propia, producción ajena y catálogo filtrado. A continuación se analiza un caso relevante en cada uno de ellos, que tenga aspectos a considerar significativos de cara al desarrollo de una herramienta paralela como VideoClipping.

Nombre del portal	HULU	Link	http://www.hulu.com
Fecha de visita	28 de Mayo de 2010		
Oferta de contenidos	Hulu ofrece contenidos audiovisuales (shows televisivos, series, películas, etc.) de importantes productoras como NBC, Fox, ABC entre otras. En su catálogo de videos ofrecen series, películas, documentales y espectáculos televisivos. Actualmente, únicamente está disponible en los Estados Unidos, evitando cualquier acceso desde otra ubicación a través de un filtro de IP. Los contenidos son completos, y únicamente se muestra la publicidad al inicio de la reproducción. Si por algo destaca esta página es por el volumen de un catálogo de calidad diverso, con producciones americanas que fácilmente pueden encontrarse en la televisión convencional.		
Temáticas	Acción y aventuras, Animación y dibujos, Arte y Cultura, Comedia, Drama, Familia, Alimentación y Ocio, Casa y Jardín, Terror y Suspense, Música, Noticias e Información, Ciencia Ficción, Deportes, Entrevistas, Videojuegos, Web y Otros.		
Lanzamiento	En Marzo de 2007 se anunció el lanzamiento de la plataforma, que usaría como partners de distribución a Facebook, MSN, AOL y Yahoo. En Agosto de ese mismo año la página se hizo pública, junto con el nombre *Hulu* ("grabación interactiva" en Chino Mandarín), pero el catálogo audiovisual no se hizo público hasta Marzo del 2008, fecha oficial de salida del portal.		
Datos adicionales	Hulu está adaptado para poder ser visto en televisiones con acceso a Internet, o mediante dispositivos como Wii, Xbox 360, PlayStation 3. Además, dispone de Plug-Ins para ser visto a través de interfaces gráficas de *MediaCenters* como PlayOn o Flex, plataformas de amplia extensión entre la comunidad usuaria de este tipo de interfaces, y ofrece un fácil manejo del reproductor en todas ellas.		

Estadísticas de acceso

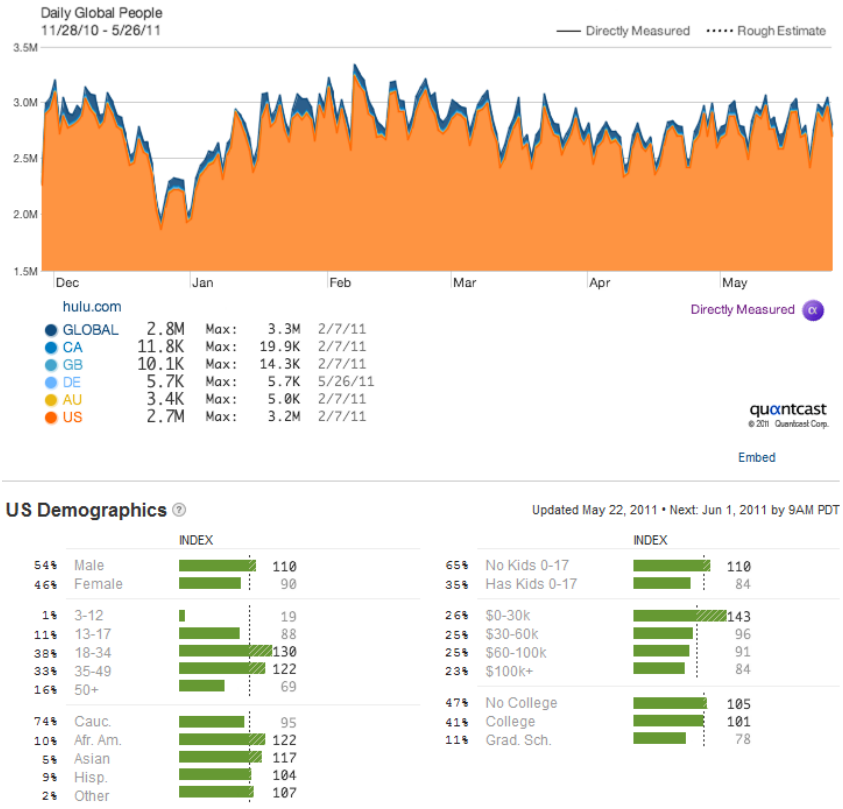


Ilustración 9: Estadísticas de Acceso Hulu. (Fuente: Quantcast - www.quantcast.com - Acceso web Global vs. EEUU)

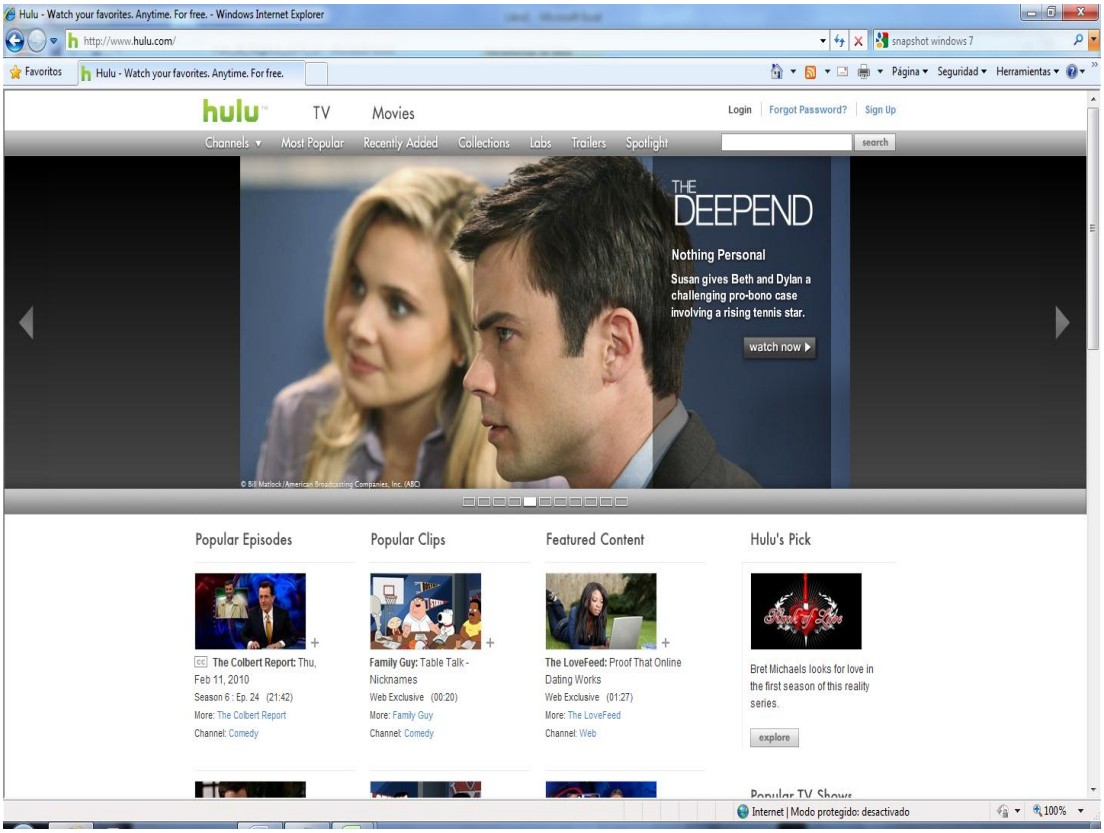
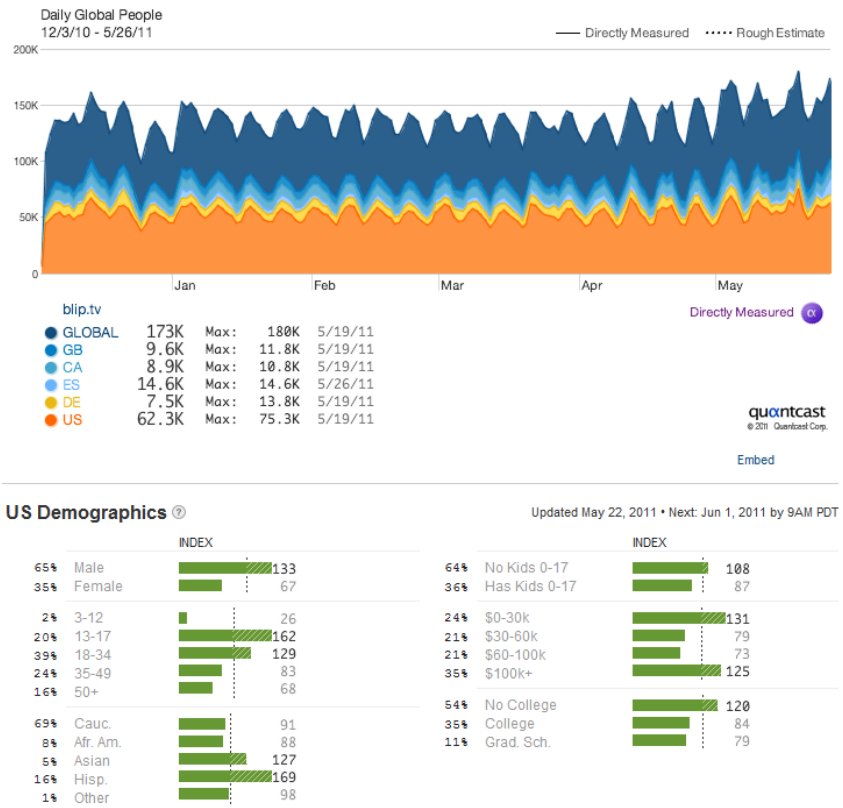


Ilustración 10: Captura de página principal Hulu.

Ilustración 11: Captura en reproducción Hulu.

Tabla 8: Tabla de análisis Hulu.

Nombre del portal	BLIP.TV	Link	http://www.blip.tv
Fecha de visita	28 de Mayo de 2010		
Oferta de contenidos	Blip TV se encuentra en las tipologías de portal de contenido no auto producido, pero no se trata de un simple contenedor de contenido audiovisual. Esta plataforma recibe vídeos de parte de usuarios que producen y editan el contenido, y BLIP.TV se encarga de filtrar y publicar. Se asigna a cada usuario un espacio en forma de Blog donde puede organizar y clasificar sus videos, además de publicar información adicional sobre ellos. Los beneficios obtenidos en publicidad previa a cada video son repartidos a medias entre la administración del portal y el propio usuario. De este modo ambas partes salen ganando: el portal dispone de videos de calidad producidos por el propio usuario (además de obtener la mitad de los ingresos publicitarios), y el usuario ve recompensado su trabajo de edición del video mediante la otra mitad de los ingresos obtenidos a través de los anuncios (tanto en los segundos previos a la reproducción como durante ella). Para dar una continuidad a los contenidos, y en busca de una fidelización del visitante, el contenido audiovisual aportado por los usuarios ha de estar editado en forma de capítulos, que son publicados con una cierta frecuencia.		

Temáticas	El abanico de temáticas es muy amplio, pero el contenido audiovisual no se encuentra organizado por ningún tipo de clasificación.
Lanzamiento	Mayo de 2005
Datos adicionales	Los contenidos del portal pueden descargarse sin ningún problema, y los derechos son cedidos al portal mientras el video se encuentre publicado en el mismo. El usuario puede cancelar la publicación de un video y borrar el fichero del servidor, recuperando los derechos. La publicidad sobre cada vídeo es opcional, de modo que el usuario la puede desactivar si así lo desea (como es obvio, en caso de realizar esta acción dejaría de percibir beneficio económico por publicidad).
Estadísticas de acceso	 Ilustración 12: Estadísticas de Acceso BlipTV. (Fuente: Quantcast - www.quantcast.com - Acceso web Global vs. EEUU)

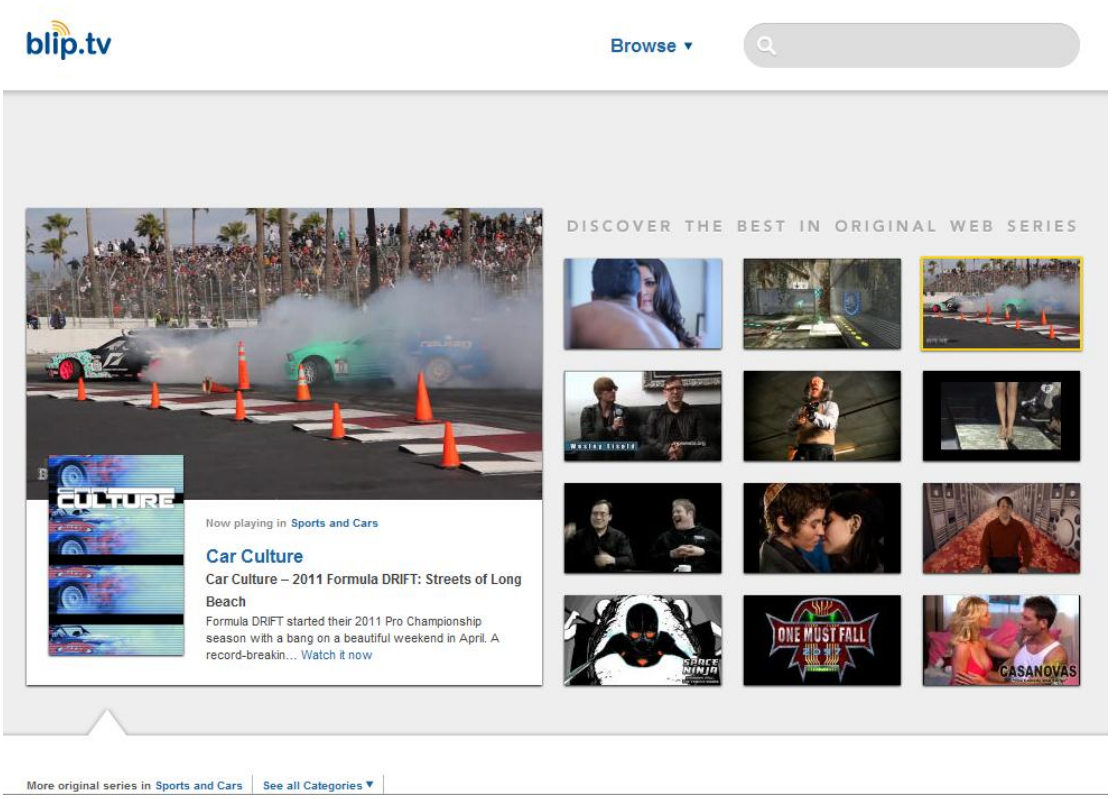


Ilustración 13: Captura de página principal BlipTV.

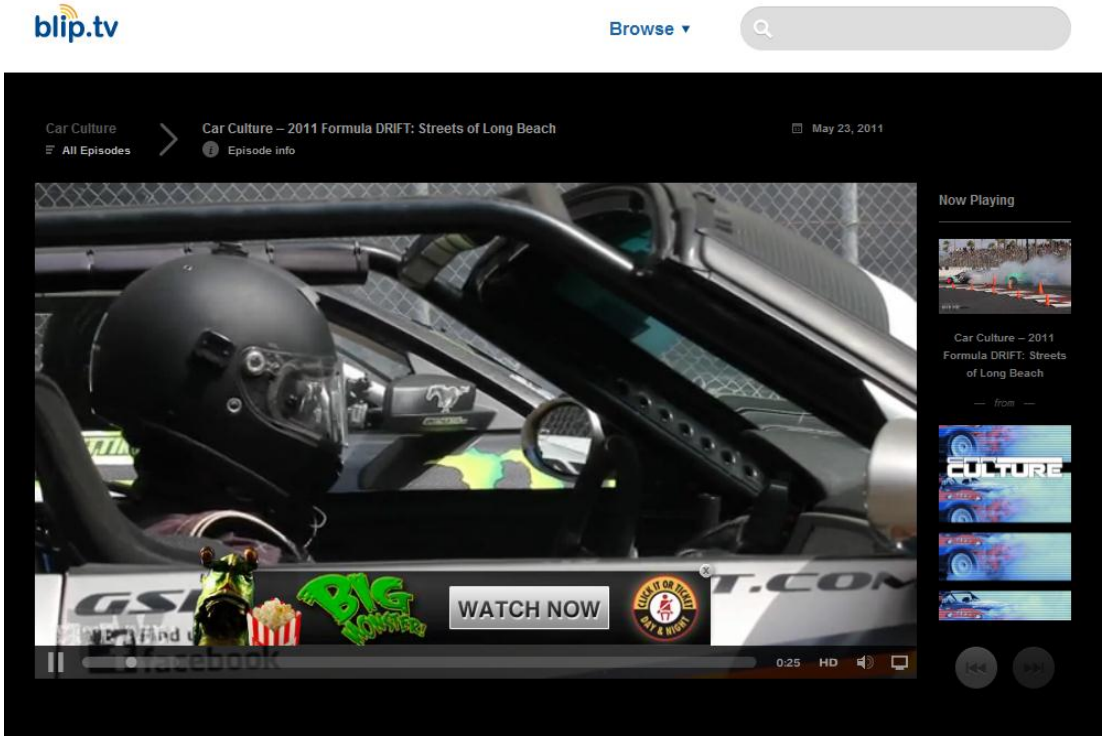
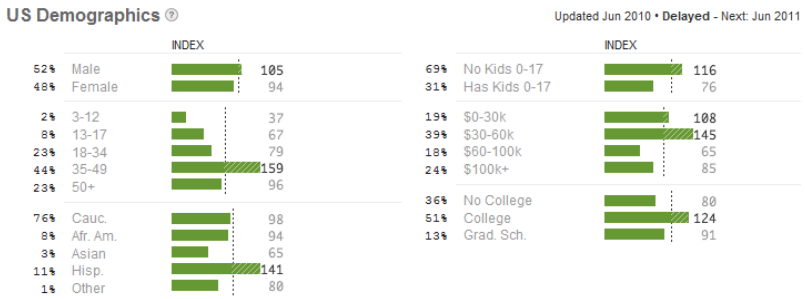


Ilustración 14: Captura en reproducción BlipTV.

Tabla 9: Tabla de análisis BlipTV.

Nombre del portal	**BEET.TV**	Link	http://www.beet.tv
Fecha de visita	28 de Mayo de 2010		
Oferta de contenidos	BEET.TV destaca entre las webs analizadas ofrecer en el catálogo de videos material de producción propia. De estructura similar a un blog, ofrece contenidos en forma de entrevistas acerca de nuevas tecnologías, a través de entrevistas a altos ejecutivos y personal especializado, incluyendo a personal de Google, Microsoft, The New York Times o Adobe. Es por tanto una web de actualidad del sector tecnológico, referente en este campo, con entradas nuevas a nivel diario.		
Temáticas	Sector tecnológico		
Lanzamiento	Mayo de 2005		
Datos adicionales	En algunos foros se cataloga a Beet.tv como blog B2B de referencia en el sector tecnológico, y se hace referencia a sus contenidos o se incrustan sus videos en importantes webs especializados: CNET News o The New York Times.		
Estadísticas de acceso	 Ilustración 15: Estadísticas de Acceso Beet.TV. (Fuente: Quantcast - www.quantcast.com - Acceso web Global vs. EEUU)		

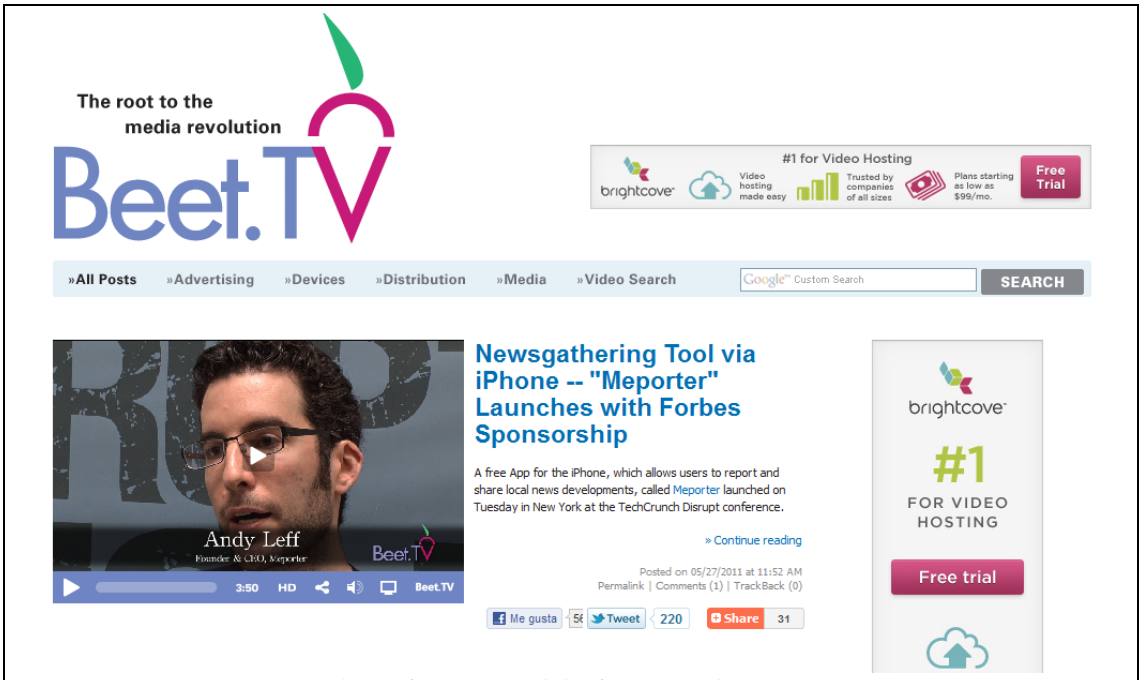


Ilustración 16: Captura de la página principal. Beet.TV.

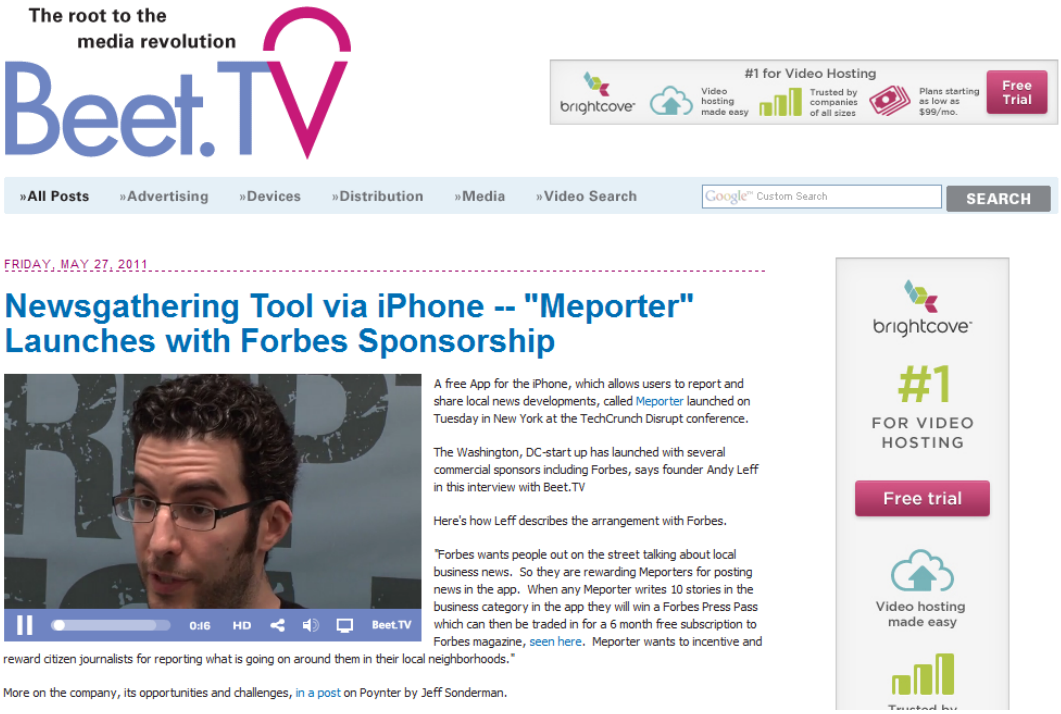
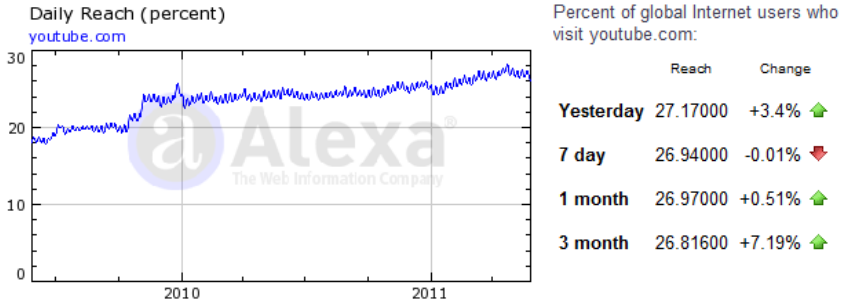
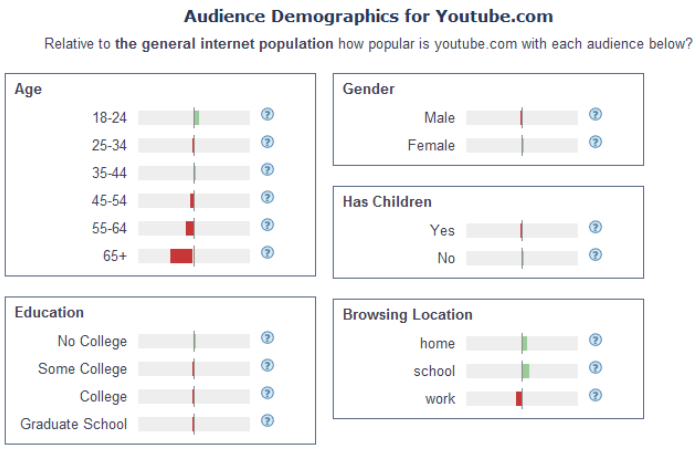


Ilustración 17: Captura en reproducción. Beet.TV.

Tabla 10: Tabla de análisis Beet.TV

Nombre del portal	YOUTUBE	Link	http://www.YouTube.com/
Fecha de visita	29 de Mayo de 2010		
Oferta de contenidos	YouTube se encuentra dentro de la tipología de portal que aloja material no autoproducido. Cualquier usuario puede visualizar contenido, pero sólo los usuarios registrados pueden cargar contenido. El número de clips que puede compartir un usuario es ilimitado, pero cada vídeo podrá tener una duración máxima de 15 minutos y un tamaño no superior a los 2GB[12]. Mediante los títulos, descripciones y etiquetas de metadatos que los usuarios asignan a sus cargas, la localización de vídeos en YouTube se hace relativamente sencilla, y aún más después de la adquisición del portal por Google en noviembre del 2006, momento a partir del cual empezaron a aparecer enlaces a los clips en las búsquedas de Google. Además de alojar vídeos de usuarios anónimos, YouTube sirve de plataforma publicitaria (para productos, artistas, empresas, etc,) como complemento a la televisión (mediante la publicación de episodios breves), para comunicaciones oficiales (Tony Blair colgó en YouTube la felicitación a Nicolas Sarkozy cuando resultó elegido), entre otros. YouTube se ha enfrentado a varias denuncias por alojar contenido con *copyright*. Para facilitar la localización del mismo ha firmado acuerdos con discográficas (Universal music y CBC directamente, y con Sony BMG y Warner a través de Google).		
Temáticas	El abanico de temáticas es muy variado y diverso. Los usuarios pueden cargar todo tipo de contenido siempre que cumpla las condiciones de uso del portal.		
Lanzamiento	Febrero de 2005		
Datos adicionales	YouTube pone al alcance de los usuarios una herramienta para denunciar infracciones en el incumplimiento de la condiciones de uso, a través de la cual pueden enviar sus críticas, comentarios y una solicitud de revisión del vídeo a los administradores del portal. Ofrece un conjunto de librerías de carácter gratuito que permiten realizar operaciones de consulta sobre el catálogo de videos de YouTube de forma externa para obtener listados de videos en función de parámetros de búsqueda, así como la obtención de elementos asociados a los videos, como entradas de comentarios.		

[12] Si la carga se hace des del mismo portal, llegando a los 20GB cuando la carga sea a través de un cargador Avanzado basado en Java. (http://en.wikipedia.org/wiki/YouTube)

	Una de las características que ofrece el portal es el subtitulado los vídeos, tanto de forma manual como de forma automática (aunque esta última no se encuentra aún disponible en todos los idiomas). Este aspecto es de vital importancia para el desarrollo de herramientas paralelas como VideoClipping y un elemento diferenciador claro con respecto a los otros portales. A nivel de expansión, YouTube ha sido capaz de captar usuarios a nivel global. Está disponible en 34 idiomas, hecho que ha propiciado que algunos usuarios lo utilicen como herramienta de denuncia social y por ello ha sido censurada en algunos países.
Estadísticas de acceso	 Ilustración 18: Estadísticas de Acceso YouTube. (Fuente: Alexa - www.alexa.com – Porcentaje de usuarios globales de Internet que acceden a YouTube)  Ilustración 19: Estadísticas de Acceso YouTube. (Fuente: Alexa - www.alexa.com – Numero medio de páginas que visita cada usuario)  Ilustración 20: Estadísticas de Acceso YouTube. (Fuente: Alexa - www.alexa.com – Características de la audiencia en comparación a los usuarios de Internet).

Visitors by Country for Youtube.com

Country	Percent of Site Traffic
United States	21.8%
Japan	6.1%
India	5.4%
Germany	4.3%
United Kingdom	3.7%
Brazil	3.6%
Italy	3.3%
Mexico	3.2%
Russia	3.0%
France	2.9%

Ilustración 21: Estadísticas de Acceso YouTube. (Fuente: Alexa - www.alexa.com – Porcentaje de usuarios por país).

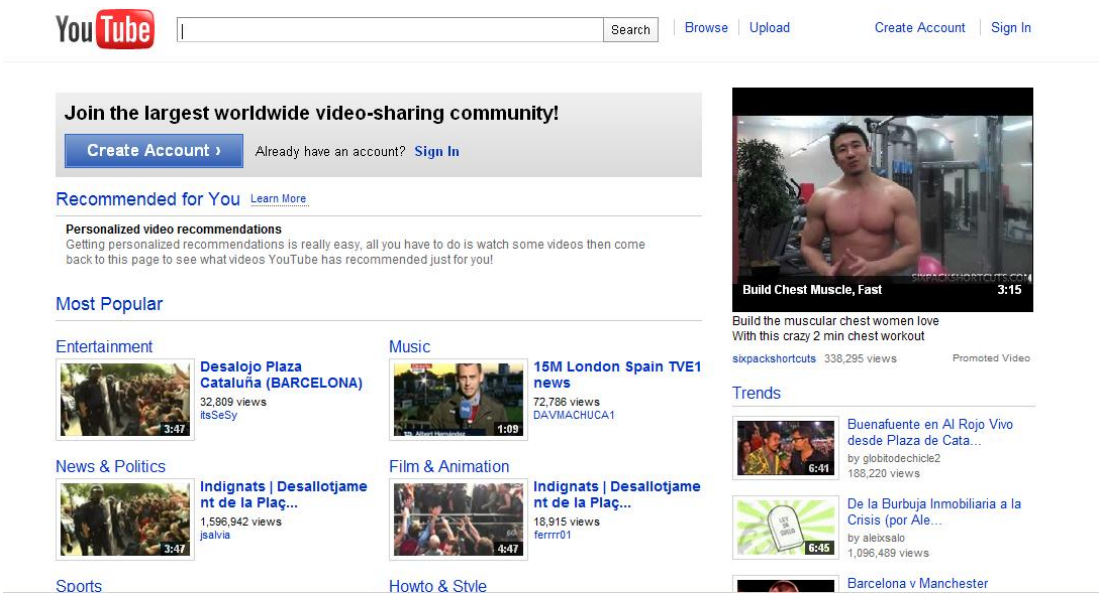


Ilustración 22: Captura de página principal YouTube.

Ilustración 23: Captura en reproducción YouTube.

Tabla 11: Tabla de análisis de YouTube.

Una vez analizadas las características que ofrecen los diferentes portales, es necesario analizar las ventajas que ofrecen los contenidos ofrecidos en cada uno de ellos en relación a las necesidades del proyecto VideoClipping.

Desde este punto de vista, la posibilidad de trabajar con elementos como los subtítulos es sin duda un elemento diferenciador de entre los otros repositorios y muy relevante para el diseño de la aplicación de anàlisis de este proyecto.

Por otro lado, la existencia de APIs de desarrollo que facilitan la realización de consultas al servidor de videos de forma externa a la propia página es también un aspecto muy importante, ya que permite poder recuperar y analizar los resultados obtenidos de forma local.

Por todo ello, y por la expansión a nivel territorial y en cuanto a volumen y calidad de contenidos, se ha considerado que YouTube es el portal repositorio de videos sobre el que centrar los esfuerzos de investigación y análisis, y sobre el que enfocar el diseño general de la aplicación VideoClipping.

4.2 YouTube API

Una vez analizado el entorno de portales de repositorios de videos, y seleccionado el portal de YouTube por contener elementos de subtítulos de gran utilidad para el desarrollo de este proyecto, se analizan a continuación las posibilidades de interacción con este portal.

YouTube, integrado en el conjunto de productos de Google, pone a disposición de los desarrolladores un Framework de herramientas con diferentes métodos de interacción con su tecnología. Entre estas herramientas destaca la presencia de una API de datos, un conjunto de librerías que permiten realizar consultas al diferente contenido disponible en las bases de datos de YouTube.

En los próximos apartados se realiza una descripción breve de los diferentes elementos disponibles en el Framework de YouTube, así como una descripción detallada del funcionamiento de la API de datos.

4.2.1 Descripción del Framework

YouTube ofrece a los usuarios la posibilidad de interactuar con sus datos a través de un conjunto de librerías "API" con las que poder consultar datos o hasta reproducir en sus propias páginas videos subidos a youtube.

A modo de resumen, las herramientas con las que permite trabajar la API de YouTube son las siguientes:

- **API de datos**: Es el componente más interesante para la realización de este proyecto, ya que permite la búsqueda de videos de forma remota, así como la parametrización de las búsquedas y la interacción con otros elementos dependientes del video a través de su identificador. Esta herramienta permite además poder interactuar con la cuenta del usuario, encapsulando todas las tareas de autentificación y haciendo posible la carga de videos al servidor de YouTube de forma remota.

- **API del reproductor**: Este elemento permite integrar el reproductor de videos de YouTube en el propio sitio web, facilitando la reproducción de videos de forma remota. Además permite crear controles personalizados que se adapten a la interfaz visual del sitio y configurar las opciones básicas de este reproductor.

- **Reproductor personalizado**: Permite integrar en el sitio web un reproductor personalizado que va más allá de la simple integración del reproductor de youtube. Se pueden configurar las opciones de visualización de los videos favoritos del usuario o ver listas de reproducción del propio usuario.

- **Widgets**: Se trata de fragmentos de código Javascript que enlazan con la página de YouTube para insertar en las webs elementos funcionales de este contenedor de videos. Entre las funcionalidades de estos elementos destacan la posibilidad de buscar videos o añadir una barra de videos como componente web.

- **YouTube direct**: Esta herramienta hace posible la interacción de los usuarios con el sitio web del usuario, añadiendo funcionalidades de envío de videos y personalización de los componentes. Además, permite administrar los videos del sitio a través de un panel de administración (llamada "Consola de moderación") desde la que el usuario puede aprovar y rechazar tanto los videos subidos por los usuarios como los comentarios asociados.

- **Panel de desarrollador**: Esta herramienta permite administrar el número de solicitudes hechas a YouTube a través de la API, así como monitorizar los errores que da la aplicación desarrollada con respecto a la API. Para este tipo de trazabilidad es necesario enviar un código de desarrollador de YouTube en cada consulta.

Como se ha comentado, de todos los elementos disponibles nos centraremos en la API de datos, que nos permite recuperar información relativa a los videos: búsquedas por palabras claves, por categoría, búsqueda de elementos asociados a los videos (como los comentarios, subtítulos, etc.), entre otros.

Por otro lado, añadir que para algunas de las funcionalidades YouTube requiere que se le envíe la clave de desarrollador en cada consulta. Este procedimiento permite por un lado verificar que las consultas realizadas de forma intensiva a los servidores de YouTube no han de ser interrumpidas por el firewall (puesto que se trata de consultas que tienen el consentimiento de esta plataforma) y por otro lado monitorizar la actividad de nuestra aplicación mediante los paneles de administración que nos ofrece.

La API de datos permite la búsqueda e interacción (subida de videos, autentificación de usuarios, etc.) a través de aplicaciones web o a través de aplicaciones de escritorio, mediante un conjunto de peticiones y respuestas por parte del servidor. Esta comunicación se puede llevar a cabo de forma directa a través de peticiones HTTP y respuestas en forma de fichero XML, o utilizando bibliotecas ya desarrolladas para diferentes lenguajes por parte de YouTube: PHP, Java, .NET o Python entre otros.

El equipo de YouTube, a través de la página de código de Google, pone a disposición de los desarrolladores la documentación necesaria para entender los procesos de comunicación con la API de datos.

4.2.2 Protocolo de la API de datos

La explicación de la API se divide en grandes bloques en base a la funcionalidad que se quiera utilizar. A modo de resumen, los bloques que trata son los siguientes:

- **Autenticación**: Explica los diferentes mecanismos de autenticación que existen para comunicarse con la API de datos. Hay que tener en cuenta que algunas de las operaciones que se pueden ejecutar como la carga de videos a YouTube o el acceso a datos de la cuenta del usuario requieren una verificación del acceso. Otras acciones como la búsqueda de videos o contenido por categoría, o contenido relacionado con los videos, no requieren autenticación con el usuario, como será nuestro caso.

- **Entradas de Videos**: Esta sección detalla la forma en la que se han de recuperar los datos de los videos, tanto de un video en concreto como la lista de un conjunto de videos a partir de una búsqueda. El formato de respuesta de una petición de video es XML, por lo que se explican las diferentes alternativas de respuesta de XML de que se disponen y se proponen ejemplos de consulta. Además, en caso de querer modificar información relativa a un video, se explican también los pasos de cómo hacerlo.

- **Recuperación y Búsqueda de videos**: Se explican las diferentes formas de consulta de videos que pone a disposición la API de datos. Esto es, búsqueda por un conjunto de palabras clave, búsqueda por una categoría o incluso búsqueda a través de listas ya generadas como los videos más vistos o los que han recibido más información por parte de los usuarios.

  Además se explican tipos de consultas relacionadas con usuarios concretos, como los vídeos subidos por un usuario o fuente concreta, como un canal de videos.

- **Subida**, **actualización y borrado de videos**: Se explican las operaciones relativas a la carga de videos en el servidor de YouTube, así como a la modificación y borrado de contenidos.

- **Funciones de comunidad**: Se explican las formas con las que se puede interactuar los contenidos relacionados con el video, como son los comentarios o la asignación de puntuaciones, entre otras acciones.

- **Otras opciones**: Se explican acciones el almacenamiento de videos o la modificación del perfil de usuario que quedan fuera del alcance de este proyecto.

Dicho esto, mencionar que las versiones de la API de YouTube tienen actualizaciones, por lo que es conveniente revisar las consultas a realizar para ejecutarlas sobre la versión adecuada. En la ejecución de las diferentes consultas se puede siempre determinar el tipo de función en base a la versión de la API a la que llamar.

4.2.3 Tipos de datos de filtrado

La API de Datos de YouTube permite definir un conjunto concreto de filtros para parametrizar las consultas a realizar. Este filtro resulta de vital importancia para el diseño de VideoClipping, al delimitar el conjunto de resultados a obtener en cada consulta y por tanto reducir drásticamente el volumen de datos a analizar.

A continuación se exponen los diferentes parámetros disponibles para estos filtros, así como la descripción del tipo de datos para cada uno de ellos.

Atributo	Descripción
caption	Epecifica si los videos han de contener subtítulos o no. Si el atributo aparece en la consulta se filtra la búsqueda por subtítulos.
category	Listado de categorías filtrable, definido por un conjunto finito de categorías con el nombre de las mismas en inglés, y con "alias" en los diferentes idiomas disponibles en YouTube (fuente: http://gdata.YouTube.com/schemas/2007/categories.cat).
format	Formato en que han de estar visibles los videos. Valores posibles: 1 (url de streaming compatible con dispositivos móbiles en formato H263), 5 (videos insertables en otros sitios) y 6 (videos compatibles para dispositivos móviles en formato MPEG4)
location	Parámetro para filtrar aquellos videos con información disponible relativa a la localización de los mismos. Formato de latitud y longitud.
location-radius	En combinación con el parámetro location, sirve para delimitar el área geográfica en la que se ubica el video. Formato de distancia en número y unidad de medida.
lr	Filtrado de videos con un título, descripción o palabras clave del mismo en un idioma concreto. En caso de busqueda de subtítulos, busca aquellos videos con subtítulos en el idioma especificado. Sintaxis de paises correspondiente a ISO 639-1 (códigos de dos dígitos indicando el idioma).
orderby	Especifica el orden en que se recuperan los videos en la búsqueda. Valores permitidos: *relevance* (relevancia), *published* (publicados), *viewCount* (contador de visualizaciones), *rating* (valoración). En caso de no indicar valor de este parámetro en la búsqueda, el filtro predeterminado es relevance.
q	Define los términos en base a los cuales se realiza la búsqueda. **YouTube busca estos términos en los títulos, las palabras clave, las descripciones, los nombres de usuario de los autores y las categorías**, no por el resto de elementos. Los términos literales (combinaciones de palabras) se incluyen entre comillas dobles y se pueden utilizar expresiones lógicas para su búsqueda.
safeSearch	Tipo de búsqueda en función de restricción de los contenidos. Valores posibles: *none* (no se aplica filtro), *moderate* (valor por defecto, filtra contenido restringido en función de la región pero no aquél que pudiera derivar a otro contenido restirngido), *strict* (se filtra todo el contenido restringido). Se restringe el contenido de forma automática en función de si puede contener o no material "sexualmente explícito".

time	Filtra la búsqueda en función de la fecha de publicación del video. Valores posibles: today (un día), this_week (siete días), this_month (un mes), all_time (valor por defecto que indica cualquier momento en el tiempo).
uploader	Permite filtrar por "Partners", creadores o productores de contenido original, en función de su nombre.
alt	Especifica el tipo de formato en que se recuperan los datos de la consulta. Tipos posibles: atom (valor por defecto), rss, json y json-in-script.
author	Permite filtrar por un usuario de YouTube concreto, indicando el nombre en forma de texto del autor.
callback	Junto con el valor de alt a json-in-script permite definir una función JavaScript sobre la que se retorna el resultado de la consulta.
max-results	Número máximo de resultados recuperado. El valor por defecto es 25 y el máximo es 50. En caso de querer más se combina este atributo con el start-index.
prettyprint	Booleano (true o false) que determina si el xml retornado está formateado con tabuladores para ser imprimido de forma fácil.
start-index	Los resultados se retornan en bloques, de cantidad definida en max-results. Mediante este filtro se puede indicar el bloque de resultados a recuperar.
strict	Se utiliza para que no se procesen aquellas solicitudes a la API que contienen parámetros que no son válidos. De esta forma se evita la consulta de llamadas con error en su construcción y su posterior respuesta errónea.
v	Especifica la versión de API con la que se realiza la consulta, por defecto es la versión 1.

Tabla 12: Filtros de la API de datos.

Respecto a los anteriores, añadir que se pueden establecer categorías personalizadas por el desarrollador, pero en este caso únicamente podrán ser recuperadas por él mismo. Es decir, como desarrolladores podemos asignar categorías personalizadas a los videos para realizar consultas específicas, pero estas categorías no son públicas para los otros usuarios.

4.2.4 Tipos de datos disponibles

La aplicación de los filtros vistos en el apartado anterior y la ejecución de la consulta, tiene como resultado un archivo XML que contiene información diversa acerca del video o del listado de videos de interés.

Estos campos, algunos de ellos opcionales, se definen en función de un conjunto de esquemas XML que almacenan toda la información disponible de forma pública acerca del conjunto de videos consultado.

En la siguiente figura se muestra un ejemplo de *feed* de videos recuperado al consultar la lista de videos más valorados ubicada en la siguiente ruta:

http://gdata.YouTube.com/feeds/api/standardfeeds/top_rated


```
<?xml version='1.0' encoding='UTF-8'?>
<feed
   xmlns='http://www.w3.org/2005/Atom'
   xmlns:app='http://purl.org/atom/app#'
   xmlns:media='http://search.yahoo.com/mrss/'
   xmlns:openSearch='http://a9.com/-/spec/opensearchrss/1.0/'
   xmlns:gd='http://schemas.google.com/g/2005'
   xmlns:yt='http://gdata.YouTube.com/schemas/2007'>

   <id>http://gdata.YouTube.com/feeds/api/standardfeeds/top_rated</id>
   <updated>2011-05-15T22:13:22.000-07:00</updated>
   <category scheme='http://schemas.google.com/g/2005#kind'            term='http://gdata.YouTube.com/schemas/2007#video'/>
   <title type='text'>Top Rated</title><logo>http://www.YouTube.com/img/pic_YouTubelogo_123x63.gif</logo>
   <link rel='alternate' type='text/html' href='http://www.YouTube.com/browse?s=tr'/>
   <link rel='http://schemas.google.com/g/2005#feed' type='application/atom+xml' href='http://gdata.YouTube.com/…top_rated'/>
   <link rel='http://schemas.google.com/g/2005#batch' type='application/atom+xml' href='http://gdata.YouTube.com/…/batch'/>
   <link rel='self' type='application/atom+xml' href='http://gdata.YouTube.com/…/top_rated?start-index=1&amp;max-
results=25'/>
   <link rel='next' type='application/atom+xml' href='http://gdata.YouTube.com/…/top_rated?start-index=26&amp;max-
results=25'/>
   <author>
      <name>YouTube</name>
      <uri>http://www.YouTube.com/</uri>
   </author>
   <generator version='2.0' uri='http://gdata.YouTube.com/'>YouTube data API</generator>
   <openSearch:totalResults>98</openSearch:totalResults>
   <openSearch:startIndex>1</openSearch:startIndex>
   <openSearch:itemsPerPage>25</openSearch:itemsPerPage>

   <entry>
      <id>http://gdata.YouTube.com/feeds/api/videos/dMH0bHeiRNg</id>
      <published>2006-04-06T21:30:53.000Z</published>
      <updated>2011-05-20T20:09:48.000Z</updated>
      <category scheme='http://schemas.google.com/…' term='http://gdata.YouTube.com/…/2007#video'/>
      <category scheme='http://gdata.YouTube.com/…categories.cat' term='Comedy' label='Comedia'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/keywords.cat' term='Dancing'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/keywords.cat' term='comedy'/>
      <title type='text'>Evolution of Dance - By Judson Laipply</title>
      <content type='text'>For more visit http://www.mightaswelldance.com</content>
      <link rel='alternate' type='text/html'
href='http://www.YouTube.com/watch?v=dMH0bHeiRNg&amp;feature=YouTube_gdata'/>
      <link rel='http://gdata.YouTube.com/schemas/…' type='application/atom+xml'
href='http://gdata.YouTube.com/…/responses'/>
      <link rel='http://gdata.YouTube.com/schemas/2007#video.related' type='application/atom+xml' href='http://…/related'/>
      <link rel='self' type='application/atom+xml'
href='http://gdata.YouTube.com/feeds/api/standardfeeds/top_rated/v/dMH0bHeiRNg'/>
      <author>
         <name>judsonlaipply</name>
         <uri>http://gdata.YouTube.com/feeds/api/users/judsonlaipply</uri>
      </author>
      <gd:comments>
         <gd:feedLink href='http://gdata.YouTube.com/feeds/api/videos/dMH0bHeiRNg/comments' countHint='517577'/>
      </gd:comments>
      <media:group>
         <media:category label='Comedia'
scheme='http://gdata.YouTube.com/schemas/2007/categories.cat'>Comedy</media:category>
         <media:content url='http://.../v/dMH0bHeiRNg?f=standard&amp;app=YouTube_gdata' type='application/x-shockwave-
flash' medium='video' isDefault='true' expression='full' duration='360' yt:format='5'/>
         <media:content url='rtsp://v5.cache7.c.YouTube.com/CiQLENy73wIaGwnYRKJ3bPTBdBMYDSANFE…=/0/0/0/video.3gp'
type='video/3gpp' medium='video' expression='full' duration='360' yt:format='1'/>
         <media:content url='rtsp://v3.cache8.c.YouTube.com/…/video.3gp' type='video/3gpp' medium='video' expression='full'
duration='360' yt:format='6'/>
         <media:description type='plain'>For more visit http://www.mightaswelldance.com</media:description>
         <media:keywords>Dancing, comedy</media:keywords>
         <media:player url='http://www.YouTube.com/watch?v=dMH0bHeiRNg&amp;feature=YouTube_gdata_player'/>
         <media:thumbnail url='http://i.ytimg.com/vi/dMH0bHeiRNg/0.jpg' height='240' width='320' time='00:03:00'/>
         <media:thumbnail url='http://i.ytimg.com/vi/dMH0bHeiRNg/1.jpg' height='90' width='120' time='00:01:30'/>
         <media:thumbnail url='http://i.ytimg.com/vi/dMH0bHeiRNg/2.jpg' height='90' width='120' time='00:03:00'/>
         <media:thumbnail url='http://i.ytimg.com/vi/dMH0bHeiRNg/3.jpg' height='90' width='120' time='00:04:30'/>
         <media:title type='plain'>Evolution of Dance - By Judson Laipply</media:title>
         <yt:duration seconds='360'/>
      </media:group>
      <gd:rating average='4.6820703' max='5' min='1' numRaters='810475' rel='http://schemas.google.com/g/2005#overall'/>
      <yt:statistics favoriteCount='1045538' viewCount='172755211'/>
   </entry>

   <entry>
      <id>http://gdata.YouTube.com/feeds/api/videos/uelHwf8o7_U</id>
      <published>2010-08-05T18:30:09.000Z</published>
      <updated>2011-05-20T20:01:22.000Z</updated>
```

```
      <app:control>
        <yt:state name='restricted' reasonCode='limitedSyndication'>Syndication of this video was restricted by the content
owner.</yt:state>
      </app:control>
      <category scheme='http://schemas.google.com/g/2005#kind' term='http://gdata.YouTube.com/schemas/2007#video'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/categories.cat' term='Music' label='Msica'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/keywords.cat' term='Love'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/keywords.cat' term='The'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/keywords.cat' term='Way'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/keywords.cat' term='You'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/keywords.cat' term='Lie'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/keywords.cat' term='Eminem'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/keywords.cat' term='Rihanna'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/keywords.cat' term='Recovery'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/keywords.cat' term='Megan'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/keywords.cat' term='Fox'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/keywords.cat' term='Dominic'/>
      <category scheme='http://gdata.YouTube.com/schemas/2007/keywords.cat' term='Monaghan'/>
      <title type='text'>Eminem - Love The Way You Lie ft. Rihanna</title>
      <content type='text'>Music video by Eminem performing Love The Way You Lie.  2010 Aftermath Records</content>
      <link rel='alternate' type='text/html' href='http://www.YouTube.com/watch?v=uelHwf8o7_U&amp;feature=YouTube_gdata'/>
      <link rel='http://...#video.responses' type='application/atom+xml' href='http://…/responses'/>
      <link rel='http://…#video.related' type='application/atom+xml' href='http://…/related'/><link rel='self'
type='application/atom+xml' href='http://gdata.YouTube.com/feeds/api/standardfeeds/top_rated/v/uelHwf8o7_U'/>
      <author>
        <name>EminemVEVO</name>
        <uri>http://gdata.YouTube.com/feeds/api/users/eminemvevo</uri>
      </author>
      <gd:comments>
        <gd:feedLink href='http://gdata.YouTube.com/feeds/api/videos/uelHwf8o7_U/comments' countHint='502568'/>
      </gd:comments>
      <media:group>
        <media:category label='Msica' scheme='http://gdata.YouTube.com/schemas/2007/categories.cat'>Music</media:category>
        <media:content url='http://www.YouTube.com/v/uelHwf8o7_U' type='application/x-shockwave-flash' medium='video'
isDefault='true' expression='full' duration='267' yt:format='5'/>
        <media:description type='plain'>Music video by Eminem performing Love The Way You Lie.  2010 Aftermath
Records</media:description>
        <media:keywords>Love, The, Way, You, Lie, Eminem, Rihanna, Recovery, Megan, Fox, Dominic,
Monaghan</media:keywords>
        <media:player url='http://www.YouTube.com/watch?v=uelHwf8o7_U'/>
        <media:restriction type='country' relationship='deny'>DE</media:restriction>
        <media:thumbnail url='http://…/0.jpg' height='240' width='320' time='00:02:13.500'/>
        <media:thumbnail url='http://…/1.jpg' height='90' width='120' time='00:01:06.750'/>
        <media:thumbnail url='http://…/2.jpg' height='90' width='120' time='00:02:13.500'/>
        <media:thumbnail url='http://…/3.jpg' height='90' width='120' time='00:03:20.250'/>
        <media:title type='plain'>Eminem - Love The Way You Lie ft. Rihanna</media:title>
        <yt:duration seconds='267'/>
      </media:group>
      <gd:rating average='4.8798766' max='5' min='1' numRaters='728933' rel='http://schemas…'/>
      <yt:statistics favoriteCount='787878' viewCount='335760800'/>
    </entry>

    <!-- En el resultado original había otras entradas que han sido eliminadas para este ejemplo. -->
    <!—Algunas rutas de este ejemplo han sido recortadas para facilitar la lectura -->
</feed>
```

Ilustración 24: Fichero de video.

Los resultados de las consultas pueden usar diferentes *NameSpaces* en el XML (esquemas de la estructura de los elementos del XML), que acotan pos posibles datos que se pueden recuperar en cada caso. Esto explica las diferentes etiquetas expuestas en el ejemplo anterior.

Según la documentación de Google estos esquemas se definen de la siguiente forma:

Esquema	Prefijo del espacio de nombres	URL de esquema
Formato de sindicación Atom	[Ninguno] - es el espacio de nombre predeterminado	http://www.w3.org/2005/Atom
Esquema de OpenSearch	openSearch	http://a9.com/-/spec/opensearchrss/1.0/
Media RSS	media	http://search.yahoo.com/mrss/
Esquema XML de YouTube	yt	http://gdata.YouTube.com/schemas/2007

Esquema de datos de Google Data	gd	http://schemas.google.com/g/2005
GeoRSS	georss	http://www.georss.org/georss
Lenguaje de marcado geográfico	gml	http://www.opengis.net/gml
Atom Publishing Protocol	app	http://www.w3.org/2007/app

Tabla 13: Esquemas de la API de datos.

Cada uno de los esquemas anteriores proporciona un tipo de datos distinto. La recuperación de la información del video puede contener información de todos ellos. En función del análisis a realizar se puede analizar una u otra entrada, teniendo en cuenta el tipo de datos de cada uno de ellos.

A continuación se explica cada uno de los elementos que forman los esquemas de la tabla anterior:

Atom: Información genérica del objeto, que puede ser un video, una lista de reproducción, una lista de videos favoritos de un usuario o un contacto. Atom es un tipo de formato de ficheros de sindicación.

Posibles campos de Atom como respuesta a la API de YouTube:

Atributo	Descripción
author	Contiene información sobre el autor del objeto (contiene nombre y uri).
category	Categoría a la que pertenece el objeto. Esta categoría puede venir definida por el esquema de categorías o por el esquema de palabras clave.
content	Texto que contiene una descripción del objeto. En el caso de un video puede contener lo mismo que media:description.
entry	Encapsula otra información del objeto: datos del video, comentarios, etc.
feed	Encapsula otra información del feed del objeto: id, categoría, título, etc.
generator	Indica el software que se ha utilizado para generar el feed.
id	Identifica de forma unívoca el objeto con una cadena alfanumérica.
link	Enlace del contenido relacionado con el video. En función del atributo “rel” de este elemento muestra enlaces a la ruta del elemento, las respuestas en forma de otros videos o los videos relacionados, entre otros.
logo	Ruta a una imagen descriptiva del objeto.
name	Nombre del usuario YouTube del autor.
published	Hora de creación del objeto (de la entrada).
title	Título del objeto.

updated	Hora de última actualización.
uri	Enlace relacionado con el autor.

Tabla 14: Campos Atom de respuesta a la API de YouTube.

OpenSearch: Las tecnologías OpenSearch se enmarcan dentro de los formatos de sindicación que sirven para guardar datos relativos al modo en que la información resultante de un motor de búsqueda tiene que ser publicada.

En el caso de YouTube los elementos que proporciona la API son:

Atributo	Descripción
itemsPerPage	Indica el número de elementos que han de mostrarse como resultado de la consulta. El máximo número de elementos posibles por página es de 50.
startIndex	Identifica el item inicial a mostrar del conjunto de items del resultado.
totalResults	Guarda información del total de valores aproximado de la consulta. El número máximo de este campo es 1.000.000.

Tabla 15: Elementos OpenSearch que proporciona la API de YouTube.

Se permite una paginación automática de los resultados mediante el elemento *link*. Si este elemento contiene un atributo llamado *rel* y su valor es *next*, entonces es que hay elementos por delante y se puede avanzar en la paginación. Si existe un elemento link con valor *rel* igual a *prev* es que existe paginación hacia atrás. El valor de estos elementos *link* contiene el enlace a la página anterior o siguiente según el caso. De este modo, es muy sencillo hacer paginación con los resultados, y no hay que analizar la página actual y los resultados totales.

YouTube: Este tipo de elementos son específicos de YouTube (los otros pueden estar compartidos por otras plataformas) y guardan información relativa al video concreto o información relativa al usuario que ha publicado el mismo, en función de los datos registrados en la cuenta de YouTube asociada (es necesario tener cuenta de YouTube para poder subir un video).

Atributo	Descripción
aboutMe	Muestra la información que el usuario haya entrado en el campo **AboutMe** (Sobre mí) en su perfil.
accessControl	Guarda información relativa a la capacidad de los usuarios de interactuar con los contenidos del video: valorarlo, añadir comentarios,… Tiene dos atributos: **action** (acción sobre la que se informa) y **permission** (permiso que se le da a la acción determinada, que puede ser aceptarla o denegarla).
age	Muestra la edad del usuario de YouTube registrado, que se calcula a partir de su fecha de nacimiento y la fecha actual.

books	Muestra los libros que el usuario haya entrado en su perfil.
channelStatistics	En el caso de la consulta de un canal, muestra información estadística relacionada con el mismo. La información puede ser tanto de videos subidos como de reproducciones hechas. Concretamente, los posibles valores son **commentCount** (número de comentarios), **totalUploadViewCount** (reproducciones de todos los videos del canal), **videoCount** (videos que contiene el canal), **viewCount** (veces que se ha visto la página del canal).
company	Especifica la empresa en la que el usuario trabaja o que consta en su perfil.
countHint	En el caso de recuperar información de una lista de videos, este campo guarda cuantos videos contiene la lista.
derived	Identifica los datos de una pista de subtítulos que ha sido creada a partir de la recuperación automática, interpretando el audio del video (ASR: *Automatic Speech Recognition*). En el caso que se hayan generado subtítulos con esta técnica, el valor de este elemento es **speechRecognition**.
duration	Especifica la duración del video en segundos mediante el atributo **seconds**. El valor es siempre in entero.
firstName	Muestra el nombre del usuario de YouTube en función de su perfil.
gender	muestra el género (hombre o mujer) del usuario de YouTube.
hobbies	Guarda las aficiones del usuario que constan en el perfil.
hometown	Guarda la ciudad de residencia del usuario.
incomplete	Este atributo indica a YouTube si tiene que generar de forma automática el valor de otros atributos que el usuario no ha entrado al subir el video. Estos atributos son: - **El título**: Si no hay uno entrado se pone el nombre del archivo el video. - **Las palabras clave**: Se pone de nuevo el nombre del archivo del video como palabras clave. - **Category:** Se pone como categoría la del último video subido por el usuario, y si es el primer video entonces se asigna a la categoría *People*.
lastName	Indica el apellido del usuario que ha subido el video.
location	Se muestra en forma de texto una pequeña descripción de dónde se grabó el video.
movies	Se muestran las películas favoritas que haya puesto el usuario en su perfil.

music	Se muestra la música favorita que haya puesto el usuario en su perfil.
occupation	Muestra la profesión que el usuario tenga puesta en el perfil.
position	Especifica el orden en que un video aparece, en el caso que se esté recuperando una lista de reproducción.
private	Indica si el video es privado. Un video sólo puede ser visto por el público en general si se considera público (no privado), de lo contrario sólamene lo pueden ver usuarios específicos que especifica la persona que lo sube. El tag no tiene valor, sólo si aparece ya indica que es privado, y si no aparece entonces es público.
rating	Indica la valoración que tiene el video. Esta valoración viene dada por tres atributos del elemento: - **numDislikes**: Entero que indica el número de personas a las que no les ha ustado el video. - **numLikes**: Entero que indica el número de personas a las que les ha gustado el video. - **value**: En el caso de enviar una petición para decir si nos gusta o no el video, hay que enviar este elemento con el valor "like" o "dislike" según si nos gusta o no en cada caso.
recorded	Muestra la fecha en la que se grabó el video.
relationship	Según los datos del perfil muestra si el usuario mantiene una relación.
school	Muestra la escuela del usuario según su perfil.
spam	Según el contenido del comentario, se añade este elemento cuando se identifica como spam.
state	Indica el motivo por el que un video no se puede reproducir, teniendo en cuenta los atributos siguientes: - **name:** Identifica el estado actual del video, los valores posibles son processing, restricted, deleted, rejected y failed. - **reasonCode:** Describe de forma más detallada el motivo por el cual un video no se puede mostrar. El dominio de posibles valores depende del valor del atributo name del mismo elemento. - **helpUrl:** Incluye un link a la ayuda de YouTube que proporciona más ayuda con la interpretación de este campo.
statistics	Informa de datos estadísticos del video. Este elemento sólo existe si el video ha sido visto alguna vez. Los posibles atributos dentro de este elemento son: - **viewCount:** Entero que muestra el número de veces que se ha visto el objeto, que puede ser un video o un perfil de usuario.

	- **videoWatchCount:** Número de videos que el usuario ha visto en YouTube, por tanto sólo aparece cuando el elemento statistics aparece al mostrar información de un usuario. - **subscriberCount:** Número de usuarios que se han subscrito al canal de un usuario específico. De nuevo este campo aparece al mostrar información de un usuario, no del video. - **lastWebAccess:** Fecha en la que el usuario del que se muestra información accedió a YouTube por última vez. - **favoriteCount:** Número de veces que el video ha sido añadido como favorito. - **totalUploadsViews:** En la información de un usuario, muestra el número de veces que se han visto los videos subidos por el usuario.
status	Muestra el estado del usuario de YouTube, y sólo aparece al hacer una consulta de los contactos de un usuario.
uploaded	Especifica cuándo se creó una lista de reproducción.
username	Indica el nombre de usuario de YouTube relacionado con el contenido. En función de lo que se esté recuperando la información de este campo es distinta (información del perfil, información del canal, información del video, etc).
videoid	Indica el Identificador único del video dentro de YouTube.

Tabla 16: Elementos YouTube que proporciona la API.

Media RSS: Es un esquema de XML que representa un estándar para organizar la información de elementos multimedia (videos o audios), y que diseñó originalmente Yahoo! Para organizar los resultados de las búsquedas en este buscador. Se ha extendido al ser utilizado para almacenar la información de *Podcasts*.

En el caso del api de YouTube, los elementos utilizados son los siguientes:

Atributo	Descripción
Category	Especifica la caegoría del video. Igual que en el filtrado se sigue el esquema definido de categorías. Este elemento contiene dos atributos: **label** (nombre de la categoría) y **scheme** (ruta al esquema de categorías[13]).
Content	Muestra diferente información relativa al contenido del video. Esta información se muestra mediante los atributos siguientes: - **url:** Indica la dirección del recurso a mostrar. - **type:** Texto que indica el tipo de contenido que se muestra, siguiendo la codificación MIME. Esta nomenclatura asigna un código formado por tipo/subtipo según la extenxión del archivo. Por ejemplo: text/html (para archivos html), image/jpeg (para imágenes en jpeg), application/pdf (para archivos PDF) o

[13] Más información en apartados posteriores.

	application/x-shockwave-flash (para archivos flash de extensión swf, que es el formato de los videos de YouTube). - **isDefault:** En el caso de recuperar un grupo de videos (media group) indica si el video concreto es el video predeterminado del grupo. En el caso de recuperar información de un video de YouTube (como es el caso) este valor, booleano, tiene valor "true". - **expression:** Texto que indica si el video es sólo un fragmento ("sample"), si está completo ("full") o si es un video *streaming* ("nonstop"). - **duration:** Es un entero que muestra la duración del video en segundos. **yt:format:** Muestra información relativa al tipo de video del objeto descrito en este nodo content. El tipo de video está codificado como número, y los posibles valores son: - **1**: Ruta RTSP para video en Straming para reproducción en móviles. Video en formato H.263 con audio AMR. - **5**: Ruta HTTP para poder incrustar el swf, y por lo tanto no compatible con móviles. - **6**: Ruta RTSP para video en Straming para reproducción en móviles. En este caso el formato del Video es MPEG-4 y audio AAC.
credit	Muestra información relativa al propietario del video o de la lista de reproducción. Se muestra también información del rol (mediante al atributo "role") del usuario con respecto al video, típicamente tiene el valor "uploader", indicando que es quien ha subido el video.
description	Es una descripción en forma de texto del contenido del video. Este campo es obligatorio en el momento de subir un video a YouTube. Además, la descripción (según la documentación de YouTube) **ha de estar hecha mediante frases, no con una serie de palabras claves**. El tamaño máximo de este campo es de 5000 bytes y la codificación es UTF-8, aunque puede contener los símbolos "<" y ">". En la recuperación de este campo mediante la API, l**os datos se limitan a 500 caracteres** (excepto si se es prpietario del video consultado). El atributo **type** indica el tipo de texto de la descripción, que en el caso de los videos de YouTube siempre tiene el valor "plain".
group	Es un elemento que sirve para agrupar otros elementos de media explicados en esta misma tabla, de modo que agrupa características de un video.
keywords	Son las palabras claves asociadas al video. Es un campo obligatorio para subir el video a YouTube, de modo que todos los videos tienen por lo menos una palabra clave. Las palabras clave o conceptos clave han de ser por lo menos de 2 bytes (2 caracteres) y no más de 30, y se han de separar por comas. En caso de tener una "palabra clave" compuesta por dos términos, hay que separarlos por espacios. Por ejemplo: *concierto, Sant Jordi, Barcelona* son 3 palabras clave, mientras que *concierto, Sant, Jordi, Barcelona* son 4 palabras clave.

	Además, el tamaño de este campo está limitado a 500 bytes, teniendo en cuena que si la palabra clave contiene espacios (porque son dos términos, como en el ejemplo de *Sant Jordi*), entonces YouTube le añade dos comillas dobles, una a cada extremo, que cuentan en el cómputo del tamaño. Es decir, que *Sant Jordi* se contaría como 12 caracteres (9 letras de los términos + 1 espacio + 2 comillas sobles).
player	Indica la url de la página que contiene el reproductor con el video para ser reproducido en la web. Esto es, reproductor con el video cargado.
rating	Muestra si hayalgún tipo de restricción en el contenido del video en el país que se está visualizando. Sólo hay esta estiqueta si hay algún tipo de restricción. Contiene el atributo "country" que especifica el pais en el que afecta la restricción. Cada país se indica mediante un código de dos letras, siguiendo la codificación ISO 3166[14]. En caso de haber más de uno, se separan por comas. Si el valor es "all" significa que hay algún tipo de restricción en todos los paises. En la documentación no se especifican posibles valores de este atributo.
restriction	Muestra si hay algun tipo de restricción con respecto a la reproducción del video, es decir, si está prohibida su reproducción en algún país. Contiene dos atributos: - type: Con valor de "Country" indicando que la restricción es a nivel de país. - relationship: indica el tipo de restricción, que puede ser *allow*o *deny*, según si se permite la reproducción o no respectivamente. Por otro lado, el valor del elemento contiene el código del país en el que se aplica la resticción, en formato de ISO 3166 comentado anteriormente.
thumbnail	Guarda información relativa a una imagen que puede ser utilizada para representar el video (un fotograma del mismo). Este elemento guarda propiedades como la url, el tamaño, el instante del video al que pertenece la imagen y el nombre de la imagen.
title	Muestra el título del vídeo, con un máximo de 100 bytes de longitud.

Tabla 17: Elementos Media RSS que proporciona la API de YouTube.

GData: Es un esquema que permite interactuar con la API de datos, concretamente con los datos recibidos por la misma. A la API de Datos de Google se la conoce también por el nombre de GData, de ahí el nombre de este esquema.

Dicho esto, los atributos con los que se trabaja en la recuperación de datos de YouTube son los siguientes:

[14] Más información acerca de esta codificación en apartados posteriores.

Atributo	Descripción
comments	Guarda todos los datos relacionados con los comentarios del video. Encapsula otros elementos.
feedLink	Guarda la ruta de un archivo de sindicación (*feed*) que contiene entradas con los comentarios relacionados con el video del que se recupera información. Además de la ruta del enlace, contiene el atributo **countHint**, con información relativa al número de comentarios en el feed con el que enlaza.

Tabla 18: Elementos GData que proporciona la API de Youube.

GML: Se trata de un esquema para compartir datos relativos a la localización, que en el caso de YouTube guarda información de la situación en que un video fue grabado.

Los campos usados son:

Atributo	Descripción
Point	Guarda información relativa a la localización del video. En caso de querer actualizar el video, también hay que actualizar la localización, de lo contrario este campo se borra.
pos	Incluye los datos de posición, en coordenadas, de la localización del video.

Tabla 19: Elementos GML que proporciona la API de YouTube.

Atom Publishing Protocol: Guarda información relativa al estado de publicación de un Video.

Los campos usados de este esquema son los siguientes:

Atributo	Descripción
control	Describe en forma de texto el estado de publicación de un video. Esta descripción puede ser que el vídeo no se ha publicado por algún error en la carga, que YouTube lo ha rechazado, que tiene la publicación restringida por algún motivo, etc. No se publica en la documentación un listado de los posibles valores de este campo.
draft	Sólo disponible para los videos que se han dejado de publicar, indica que ha sido quitado de los videos públicos.
edited	Indica cuándo hubo un cambio en la publicación del video por última vez. No siempre que hay elementos de Atom Publishing Control en los videos, como **control**, existe el elemento **edited**.

Tabla 20: Elementos Atom Publising Protocos que usa la API de YouTube.

Finalmente, y aunque no se dispone de un esquema público de la estructura y de los elementos soportados, existen un conjunto de elementos que sirven para describir errores en las llamadas a la API de YouTube. Entre los posibles elementos descriptivos de error podemos destacar:

Atributo	Descripción
code	Mediante un conjunto de posibles atributos, define el motivo del error por el que YouTube no puede retornar los datos esperados. Los atributos se agrupan en: - **validation**: La consulta que se ha hecho tiene algún parámetro incorrecto y no se ha podido interpretar por la API. - **quota**: Errores relativos a haber excedido el número de peticiones en un periodo de tiempo (no se especifica un límite), o errores de espacio por querer subir un video cuando el usuario ha alcanzado su cuota máxima. - **authentication**: Errores por autenticación del usuario. - **service**: Errores por estar el servidor de YouTube en tareas de mantenimiento y encontrarse temporalmente fuera de servicio.
domain	Define el tipo de error que provoca que haya fallado la consulta, que se especifica de forma más detallada con el elemento **code**.
error	Encapsula las diferentes características del error, explicadas mediante los elementos anteriores.
errors	Encapsula diferentes *items* de error.

Tabla 21: Elementos para la descripción de errores en las llamadas a la API de YouTube.

Añadir que algunos de los elementos anteriores hacen uso de dos listados de datos o tipos de archivo que se detallan en los siguientes apartados.

Llegados a este punto, y teniendo ahora en cuenta todos los datos vistos, se puede identificar la información de la respuesta recibida en la llamada del top de videos vista anteriormente, fragmentando los elementos de la siguiente forma:

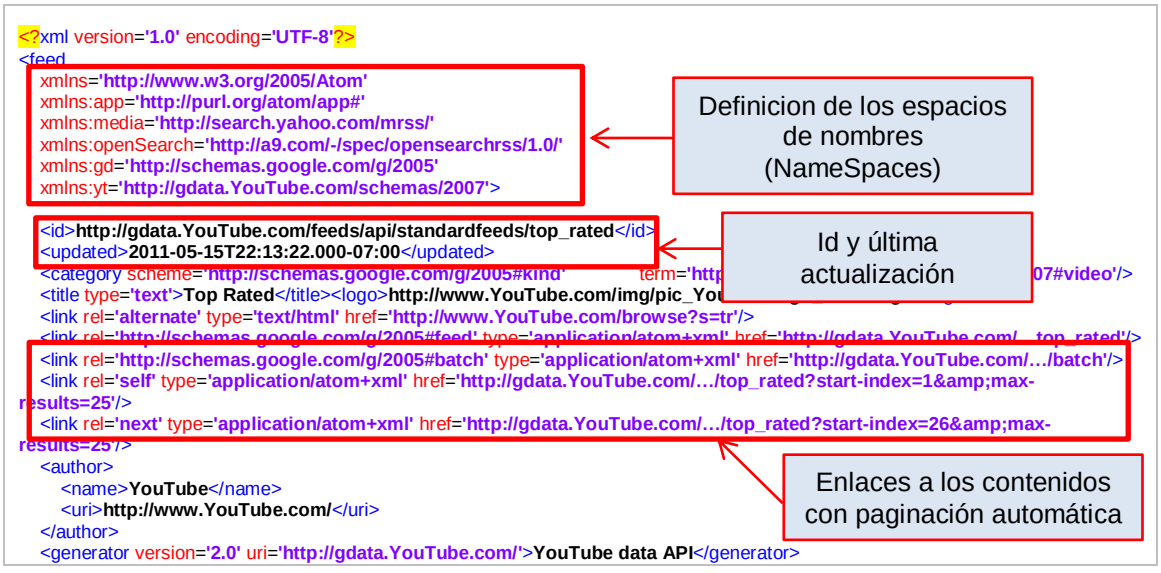


Ilustración 25: Identificación de elementos de video.

4.2.5 Categorías de Videos

Como se ha comentado a través de los apartados anteriores, los videos se catalogan en diferentes categorías. Éstas vienen definidas por un esquema público, ubicado en la siguiente ruta:

http://gdata.YouTube.com/schemas/2007/categories.cat

Este fichero, formateado mediante XML, cataloga las diferentes categorías disponibles en función de la región. A continuación se presenta un extracto del mismo:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <app:categories xmlns:app="http://www.w3.org/2007/app" xmlns:atom="http://www.w3.org/2005/Atom"
    xmlns:yt="http://gdata.youtube.com/schemas/2007" fixed="yes"
    scheme="http://gdata.youtube.com/schemas/2007/categories.cat">
  - <atom:category term="Film" label="Film & Animation" xml:lang="en-US">
      <yt:assignable />
      <yt:browsable regions="AR AU BG BR CA CZ DE DK DZ EG ES ET FI FR GB GR HK HR HU ID IE IL IN IR IT JO
        JP KR LT LV MA MX MY NL NO NZ PH PL PT RO RS RU SA SE SI SK TH TN TR TW TZ UA US VN YE ZA" />
    </atom:category>
  - <atom:category term="Autos" label="Autos & Vehicles" xml:lang="en-US">
      <yt:assignable />
      <yt:browsable regions="AR AU BG BR CA CZ DE DK DZ EG ES ET FI FR GB GR HK HR HU ID IE IL IN IR IT JO
        JP KR LT LV MA MX MY NL NO NZ PH PL PT RO RS RU SA SE SI SK TH TN TR TW TZ UA US VN YE ZA" />
    </atom:category>
  - <atom:category term="Music" label="Music" xml:lang="en-US">
      <yt:assignable />
      <yt:browsable regions="AR AU BG BR CA CZ DE DK DZ EG ES ET FI FR GB GR HK HR HU ID IE IL IN IR IT JO
        JP KR LT LV MA MX MY NL NO NZ PH PL PT RO RS RU SA SE SI SK TH TN TR TW TZ UA US VN YE ZA" />
    </atom:category>
  - <atom:category term="Animals" label="Pets & Animals" xml:lang="en-US">
      <yt:assignable />
      <yt:browsable regions="AR AU BG BR CA CZ DE DK DZ EG ES ET FI FR GB GR HK HR HU ID IE IL IN IR IT JO
        JP KR LT LV MA MX MY NL NO NZ PH PL PT RO RS RU SA SE SI SK TH TN TR TW TZ UA US VN YE ZA" />
    </atom:category>
```

Ilustración 26: Archivo de categorías en inglés.

Como se puede observar, se distribuyen las diferentes categorías mediante los nodos *category*. Estos elementos tienen los atributos:

Atributo	Descripción
term	Nombre con el que se identifica a la categoría en el momento de filtrar o recuperar los resultados.
label	Título de la categoría en un idioma específico.

En inglés estos dos campos tienen un mismo valor en alguno de los casos (por ejemplo en la categoría *Music*). Esto es debido a que el fichero de esquema mostrado cambia en función del idioma con el que esté configurado el explorador al hacer la consulta.

Es decir, en caso de tener el explorador configurado en Español, el campo “label” contiene una descripción en Español del título de la categoría. En el caso de la imagen anterior la configuración del explorador tenía el idioma Inglés por defecto. Ejecutando la misma consulta pero cambiando las propiedades del idioma por defecto la consulta tendría el siguiente resultado:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <app:categories xmlns:app="http://www.w3.org/2007/app" xmlns:atom="http://www.w3.org/2005/Atom"
    xmlns:yt="http://gdata.youtube.com/schemas/2007" fixed="yes"
    scheme="http://gdata.youtube.com/schemas/2007/categories.cat">
  - <atom:category term="Film" label="Cine y animación" xml:lang="es-ES">
      <yt:assignable />
      <yt:browsable regions="AR AU BG BR CA CZ DE DK DZ EG ES ET FI FR GB GR HK HR HU ID IE IL IN IR IT JO
        JP KR LT LV MA MX MY NL NO NZ PH PL PT RO RS RU SA SE SI SK TH TN TR TW TZ UA US VN YE ZA" />
    </atom:category>
  - <atom:category term="Autos" label="Automoción" xml:lang="es-ES">
      <yt:assignable />
      <yt:browsable regions="AR AU BG BR CA CZ DE DK DZ EG ES ET FI FR GB GR HK HR HU ID IE IL IN IR IT JO
        JP KR LT LV MA MX MY NL NO NZ PH PL PT RO RS RU SA SE SI SK TH TN TR TW TZ UA US VN YE ZA" />
    </atom:category>
  - <atom:category term="Music" label="Música" xml:lang="es-ES">
      <yt:assignable />
      <yt:browsable regions="AR AU BG BR CA CZ DE DK DZ EG ES ET FI FR GB GR HK HR HU ID IE IL IN IR IT JO
        JP KR LT LV MA MX MY NL NO NZ PH PL PT RO RS RU SA SE SI SK TH TN TR TW TZ UA US VN YE ZA" />
    </atom:category>
  - <atom:category term="Animals" label="Mascotas y animales" xml:lang="es-ES">
      <yt:assignable />
      <yt:browsable regions="AR AU BG BR CA CZ DE DK DZ EG ES ET FI FR GB GR HK HR HU ID IE IL IN IR IT JO
        JP KR LT LV MA MX MY NL NO NZ PH PL PT RO RS RU SA SE SI SK TH TN TR TW TZ UA US VN YE ZA" />
    </atom:category>
```

Ilustración 27: Archivo de categorías en español.

Como se puede observar, el campo *label* contiene en este caso la descripción en español. El atributo *lang* indica el idioma en el que aparece el atributo *label*.

Por otro lado, las categorías contienen los siguientes elementos:

yt:asignable
Indica que los nuevos videos pueden ser asignados a esta caegoría. En caso de no poder asignar nuevos videos, en lugar de este elemento tendría “yt:unassignable”

yt:browsable
Indica que la categoría se puede explorar a través de YouTube en las regiones que aparecen en el campo *regions*. El código de las regiones está formado por 2 dígitos, tal y como se explica en el próximo apartado.

4.2.6 Códigos de Paises

Respecto a los códigos de los países, como se ha podido ver están definidos mediante un código de dos dígitos. La codificación utilizada es el estándar ISO 3166 (definido por el International Organization for Standarization). El listado oficial, compuesto por 248 países, está disponible de forma pública en la siguiente ruta:

http://www.iso.org/iso/country_codes/iso_3166_code_lists/english_country_names_and_code_elements.htm

Un extracto de este listado queda representado en la figura siguiente:

Nombre del País	Codificación ISO 3166-1-alpha-2
SINGAPORE	SG
SINT MAARTEN (DUTCH PART)	SX
SLOVAKIA	SK
SLOVENIA	SI

SOLOMON ISLANDS	SB
SOMALIA	SO
SOUTH AFRICA	ZA
SOUTH GEORGIA AND THE SOUTH SANDWICH ISLANDS	GS
SPAIN	ES
SRI LANKA	LK
SUDAN	SD
SURINAME	SR
SVALBARD AND JAN MAYEN	SJ
SWAZILAND	SZ
SWEDEN	SE
SWITZERLAND	CH
SYRIAN ARAB REPUBLIC	SY
…	…

Tabla 22: Códigos de países según IDO 3166.

4.2.7 Archivo de Subtítulos

La información relativa a los subtítulos de un video no es directamente accesible a través de la API. Si bien desde la documentación oficial se informa de la posibilidad de consultar estos ficheros, no se informa del detalle de cómo realizar esta operación. Por ello, la información mostrada en este apartado ha sido consecuencia de un proceso de investigación de las direcciones y las llamadas de los elementos de YouTube, además de la búsqueda a través de foros especializados y la investigación del funcionamiento de herramientas para la edición de videos y subtítulos de YouTube.

Tras esta investigación, se ha concluido que el archivo de subtítulos relacionado con un video puede consultarse mediante una ruta URL construida conociendo el identificador del video. Como ya se ha mencionado en apartados anteriores, este identificador puede consultarse mediante la API de datos, o también puede extraerse de una dirección de video de YouTube:

http://www.youtube.com/watch?v=oaczosE4NkU

En este caso, el identificador de video sería: **oaczosE4NkU**

Una vez obtenido el identificador, se puede generar otra consulta con la que recuperar un listado de los diferentes archivos de subtítulo disponibles para el video, identificando el idioma para cada uno de ellos. La llamada necesaria para esta consulta es la siguiente:

http://video.google.com/timedtext?type=list&v=oaczosE4NkU

Donde de nuevo el parámetro **v** establece el identificador del video a buscar. Esta llamada retorna los datos:

```
<?xml version="1.0" encoding="utf-8" ?>
- <transcript_list docid="-6798408338373069243">
    <track id="0" name="" lang_code="es" lang_original="Español" lang_translated="Spanish" />
  </transcript_list>
```

Ilustración 28: Archivo de subtítulos disponibles.

En el caso de este video, únicamente se puede disponer del archivo de subtítulos en Español. Utilizando el parámetro **lang_code** de este archivo resultante, se puede generar una nueva llamada a la siguiente ruta:

http://video.google.com/timedtext?type=track&v=oaczosE4NkU&lang=es

El resultado de la llamada a esta URL queda reflejado en la figura siguiente:

```
<?xml version="1.0" encoding="utf-8" ?>
- <transcript>
    <text start="0.75" dur="5.25">Un equipo de fútbol fue el centro del mundo, ¿y cuál es el secreto? – se preguntaban todos.-</text>
    <text start="6.25" dur="4.75">¿El fútbol más bonito? ¿la velocidad? ¿el control? ¿ir siempre al ataque?</text>
    <text start="11" dur="6.006">¿buscar siempre el gol? ¿el juego en equipo? ¿no será la humildad? Sin protagonistas, ni ninguna divinidad</text>
    <text start="17.123" dur="3.877">Un equipo de fútbol fue el centro del mundo, ¿y cuál es el secreto? – se preguntaban todos.-</text>
    <text start="21.075" dur="4.325">El trabajo bien hecho, ¿quién no lo recuerda? Algo muy nuestro en los momentos de mayor gloria.</text>
    <text start="25.401" dur="4.099">Apreciar tu oficio, vigilar cada detalle, cantó un poema Joan Maragall.</text>
    <text start="29.501" dur="4.036">Está todo por hacer, podemos alzar el vuelo. Todo es posible, escribía Martí i Pol.</text>
    <text start="33.548" dur="4.452">Salvador Dalí, Pau Casals, Antoni Gaudí, desde aquí, universales.</text>
    <text start="38.11" dur="4.89">La Ruscalleda, con la cuchara, con la pluma Monzó. Tú con tu martillo, yo con mi ordenador</text>
    <text start="43.09" dur="5.41">Médicos, periodistas, mecánicos, pintores, carpinteros, fontaneros, músicos y escritores.</text>
    <text start="48.725" dur="4.275">Un equipo de fútbol fue el centro del mundo, ¿y cuál es el secreto? – se preguntaban todos.-</text>
    <text start="53.231" dur="5.269">La pasión, la dedicación, el atrevimiento, la curiosidad, y querer nuestro trabajo es nuestra creatividad.</text>
    <text start="58.883" dur="4.117">Son las ganas, trabajar duro, despierta compañero, es mejor que la suerte.</text>
    <text start="63.043" dur="2.457">No nos tenemos que reinventar, tenemos que seguir siendo quienes somos,</text>
    <text start="65.664" dur="2.336">hacer las cosas como siempre, que no nos venza el sueño.</text>
    <text start="68.427" dur="4.2">Un equipo de fútbol fue el centro del mundo, ¿y cuál es el secreto? – se preguntaban todos.-</text>
    <text start="72.632" dur="5.118">Si hacemos las cosas como nosotros sabemos, ¿hay algo imposible? ¿qué no conseguiremos?</text>
    <text start="78.6" dur="5.4">Y lo que hagan los demás da igual. El trabajo bien hecho ni tiene fronteras, ni tiene rival</text>
    <text start="86" dur="4.05">Subtítulos: Marina Giménez www.tvspotblog.com</text>
  </transcript>
```

Ilustración 29: Archivo de subtítulos.

Como se puede observar, se obtiene un XML que contiene un conjunto de entradas de texto. Estas entradas contienen, además, la marca de tiempo del inicio de la visualización del texto, expresada en segundos y milisengundos con respecto al inicio del video. Además, se dispone del parámetro **dur** que indica el tiempo (de nuevo expresado en segundos y milisegundos) durante el cual el texto de la entrada se muestra en la reproducción del video.

Es posible que el fichero de subtítulos disponible no sea el adecuado para las necesidades del sistema. En este caso, se puede también hacer una llamada al servicio de traducción automática de Google mediante la parametrización de la consulta. Los atributos a añadir son **lang**, que indica el lenguaje de origen, y **tlang**, que indica el lenguaje destino de la traducción. De este modo, la traducción de este fichero al inglés podría hacerse mediante:

http://video.google.com/timedtext?type=track&v=oaczosE4NkU&lang=es&tlang=en

El resultado de esta llamada es el siguiente:

```
<?xml version="1.0" encoding="utf-8" ?>
- <transcript>
    <text start="0.75" dur="5.25">A football team was the center of the world, and what is the secret? - Asked all .-</text>
    <text start="6.25" dur="4.75">Is the most beautiful football? Is the speed? Is the control? "Always go to the attack?</text>
    <text start="11" dur="6.006">"Always look for the goal? Does the team play? Humility is not it? Without players, or no god</text>
    <text start="17.123" dur="3.877">A football team was the center of the world, and what is the secret? - Asked all .-</text>
    <text start="21.075" dur="4.325">A job well done, who does not remember? Something in our moments of greatest glory.</text>
    <text start="25.401" dur="4.099">Appreciate your job, watch every detail, he sang a poem by Joan Maragall.</text>
    <text start="29.501" dur="4.036">It's all done, we can take flight. Anything is possible, wrote Martí i Pol</text>
    <text start="33.548" dur="4.452">Salvador Dalí, Pablo Casals, Antoni Gaudí, from here, universal.</text>
    <text start="38.11" dur="4.89">The Ruscalleda, spoon, pen Monzó. You with your hammer, me with my computer</text>
    <text start="43.09" dur="5.41">Doctors, journalists, mechanics, painters, carpenters, plumbers, musicians and writers.</text>
    <text start="48.725" dur="4.275">A football team was the center of the world, and what is the secret? - Asked all .-</text>
    <text start="53.231" dur="5.269">The passion, dedication, daring, curiosity, and love our work is our creativity.</text>
    <text start="58.883" dur="4.117">Is the desire, work hard, wake up buddy, you better luck.</text>
    <text start="63.043" dur="2.457">We do not have to reinvent, we have to remain who we are</text>
    <text start="65.664" dur="2.336">do things as usual, we do not sleep expires.</text>
    <text start="68.427" dur="4.2">A football team was the center of the world, and what is the secret? - Asked all .-</text>
    <text start="72.632" dur="5.118">If we do things as we know, is there anything impossible? What will not get?</text>
    <text start="78.6" dur="5.4">And others do not care. A job well done and has no borders, no rival has</text>
    <text start="86" dur="4.05">Subtitles: Marina Giménez www.tvspotblog.com</text>
  </transcript>
```

Ilustración 30: Archivo de subtítulos traducido al inglés.

4.2.8 Archivo de Comentarios

Teniendo en cuenta los tipos de datos recuperables de un video, el fichero de comentarios se puede obtener a partir del valor del atributo **link** recuperado de los videos, o directamente a través de:

http://gdata.youtube.com/feeds/api/videos/TrpczEs8Rfo/comments

Donde, en este caso TrpczEs8Rfo es el identificador del video. El resultado de esta llamada es un XML con información diversa de los comentarios publicados para el video.

El fichero está formado por un conjunto de entradas, que disponen de la siguiente información:

```
<entry>
        <id>http://gdata.youtube.com/feeds/api/videos/dMH0bHeiRNg/comments/Tu3ZnKUTPyElAR
VT0lgkuP0YE6y9_wEVJGqzrIZIX4k</id>
        <published>2011-05-22T17:14:23.000Z</published>
         <updated>2011-05-22T17:14:23.000Z</updated>
        <category scheme='http://…/2007#comment'/>
        <title type='text'>Haha The songs  we ...</title>
        <content type='text'>Haha The songs  we played in Music,  So much fun dancing
around on the last day of school. =) I love this video, and the music. =-)</content>
        <link rel='related' type='application/atom+xml' href='http://…/dMH0bHeiRNg'/>
        <link rel='alternate' type='text/html' href='http://…dMH0bHeiRNg'/>
        <link rel='self' type='application/atom+xml' href='http://…rIZIX4k'/>
        <author>
                <name>gabygirl747</name>
                <uri>http://gdata.youtube.com/feeds/api/users/gabygirl747</uri>
        </author>
</entry>
```

Por lo tanto, se almacena información no sólo del comentario en sí (campo **content**), sino de la fecha de publicación e incluso del autor del mismo.

4.2.9 Ejemplos de uso

Algunos de los ejemplos que se muestran en este apartado se han realizado haciendo uso del lenguaje PHP. Si bien, son extensibles a cualquier otro tipo de lenguaje de programación con soporte para la API de Datos, y el resultado obtenido en todos ellos sería equivalente al mostrado aquí.

Para hacer uso de la API de YouTube a través de PHP es necesario descargarse un Framework denominado "Zend_Framework" que contiene un conjunto de librerías para poder interactuar con YouTube. Así, por ejemplo, el código inicial de las consultas a YouTube hechas en PHP deberían iniciarse con:

```
<?php
require_once 'Zend/Loader.php';
Zend_Loader::loadClass('Zend_Gdata_YouTube');
$yt = new Zend_Gdata_YouTube();
?>
```

Este código va a buscar el archivo "Loader.php" dentro de la carpeta "Zend". Esta carpeta contiene otros archivos que sirven de enlace con la API de datos. Al crear el objeto "Zend_Gdata_YouTube" tenemos acceso al conjunto de funciones de consulta que la librería pone a nuestra disposición.

Por ejemplo, en caso de querer consultar datos relativos a un video en concreto, el código a ejecutar sería:

```
$yt->getVideoEntry('oaczosE4NkU');
```

Donde **oaczosE4NkU** sería la entrada del vídeo a consultar, el identificador del video. Como se ha comentado en apartados anteriores, este identificador se puede obtener también a través de la barra de direcciones de YouTube, consultando el parámetro "v":

```
www.youtube.com/watch?v=oaczosE4NkU
```

Ilustración 31: Página de video de YouTube.

La consulta anterior retorna una entrada de video en formato XML, que contiene campos como el título del video, la descripción, la categoría, los tags asociados al video, la duración o el contador de visitas, entre otros. El objeto necesario para la recuperación de este tipo de información es una entrada de video (*VideoEntry* en la biblioteca de PHP), que contiene diversas funciones públicas para recuperar sus atributos.

```
$entradaVideo = $yt->getVideoEntry('oaczosE4NkU');
```

A partir de aquí se podrían consultar sus campos haciendo uso de los métodos públicos:

```
$entradaVideo->getVideoTitle()→Obtenemos el Título del vídeo
$entradaVideo->getVideoDescription() → Obtenemos la descripción del video
```

Del mismo modo, se pueden hacer consultas de listas de video determinadas que son también accesibles a través de la web de YouTube, como listados de los videos más vistos, los más valorados, etc.

Además de hacer uso de una API relacionada con un lenguaje específico, es posible realizar una consulta directamente mediante una dirección web.

Este tipo de llamada, por lo que se ha podido encontrar en la documentación, es mucho más versátil y está más extendida. Permite una parametrización más sencilla de las consultas y una verificación mucho más fácil de los resultados a obtener, al dar la posibilidad de probar los resultados en el mismo navegador web. Por supuesto, todas las consultas mostradas se podrían hacer mediante los dos procedimientos, mediante la API de un lenguaje o mediante llamadas a través de la URL, y el resultado final sería el mismo.

La llamada mediante URL tiene la forma:

```
http://gdata.youtube.com/feeds/api/[parámetros]
```

En el caso de querer recuperar una lista predeterminada, entonces como parámetros sería necesario añadir una barra seguido del tipo de lista a recuperar. Hay diferentes tipos de listas predeterminadas: videos más valorados, más populares, más vistos, etc.

Así, por ejemplo, en el caso de querer consultar el listado de videos más vistos la consulta sería a la página:

```
http://gdata.youtube.com/feeds/api/standardfeeds/most_viewed
```

Como se ha mencionado anteriormente, este tipo de consulta se puede parametrizar. Estos listados, por ejemplo, se podrían filtrar por región. Recuperando el caso del ejemplo anterior, el listado de videos más vistos en un país determinado[15], por ejemplo España, sería:

```
http://gdata.youtube.com/feeds/api/standardfeeds/ES/most_viewed
```

[15] Ver el apartado *4.2.6 Códigos de Paises* de este mismo proyecto para más información acerca de los códigos de paises disponibles.

Por último, mencionar que se pueden aplicar varios parámetros por consulta en el tipo de consultas por URL, separando cada uno de ellos por barras. En el siguiente ejemplo se buscan videos con los términos “u2” y “stadium” dentro de la categoría “music”.

```
http://gdata.youtube.com/feeds/api/videos?category=music/u2/stadium
```

5 Análisis y Diseño del prototipo VideoClipping

Una vez analizadas las posibilidades que ofrece la API de datos de YouTube, se tienen estos datos en cuenta para exponer de forma detallada a continuación los diferentes requerimientos que definen el diseño general de la aplicación VideoClipping.

En primer lugar, se realiza una especificación de los requerimientos generales de la aplicación haciendo uso de un estándar de definición de requerimientos, en el que se expone el propósito del proyecto y su entorno, una descripción general de las funcionalidades y un detalle de los requerimientos.

A continuación, se analizan los casos de uso correspondientes a los diferentes usuarios y elementos que forman el sistema, y se concluye este apartado con una descripción de la arquitectura teniendo en cuenta el diseño general de la aplicación.

5.1 Especificación de requerimientos

La especificación de requerimientos que a continuación se expone se ha realizado siguiendo la norma *IEE 830-1998 – "IEEE Recommended Practice for Software Requirements Specifications"* que define la forma en la que se han de definir los requisitos de una aplicación.

La aplicación de esta norma tiene como objetivo la generación de un informe que describe el propósito general del proyecto, una descripción general del mismo y una descripción de las diferentes funcionalidades del sistema. El documento resultante de la aplicación de esta norma recibe el nombre de **SRS** (*Software Requirements Specifications*) y es el que a continuación se expone:

Tabla de contenido

5.1.1 Introducción

Este documento define el conjunto de requerimientos del software del sistema VideoClipping diseñado durante la realización del TFM. Para la redacción y estructuración del mismo se ha seguido la norma *IEE 830-1998 – "IEEE Recommended Practice for Software Requirements Specifications"*.

5.1.1.1 Propósito

El propósito de este documento es la definición de forma clara y concisa de los requerimientos del software diseñado durante la realización del TFM. Los requerimientos que aquí se definen se han basado en el cumplimiento de unos estándares de calidad definidos según la norma *ISO/IEC 9126-1Parte 1: Quality Model*, bajo unos parámetros de funcionalidad, fiabilidad, usabilidad, eficiencia, mantenimiento y portabilidad.

Este documento se dirige principalmente al equipo de desarrollo del sistema, sirviendo de guía para su posterior diseño detallado e implementación. Asimismo, también se destina a un grupo piloto de usuarios que puedan validar estos requerimientos como respuesta a unas necesidades no cubiertas por otras soluciones en cuanto al análisis de videos.

Conviene tener en cuenta que este es un proyecto de investigación, por lo que es posible que algunos de los parámetros aquí definidos se vean ligeramente alterados durante el desarrollo del sistema.

5.1.1.2 Alcance

VideoClipping es un sistema pensado para el análisis de los contenidos audiovisuales del repositorio de videos online YouTube con respecto a una determinada marca o concepto.

Basado en un conjunto de filtros iniciales a rellenar por parte del usuario mediante un formulario web, el sistema será capaz de analizar el contenido del video y los datos relacionados para obtener un video final con extractos de otros videos en los que aparece el concepto buscado o un concepto similar. De este modo, el audiovisual final será una concatenación de extractos de aquellos momentos en los que se hace referencia al término o términos buscados.

Este sistema extraerá datos de YouTube, los analizará, los procesará y obtendrá un resultado en forma de video. Sin embargo, no editará los videos ni les añadirá ningún tipo de elemento de posproducción. Tampoco cambiará el formato original de los mismos, ni en el audio ni en el video.

VideoClipping se concibe como una herramienta *online* de gran utilidad para la búsqueda de tendencias de mercado, así como para el análisis de una marca y los conceptos asociados a la misma dentro del conjunto de videos generados por la comunidad de usuarios de YouTube.

Los resultados obtenidos han de ir mucho más allá de lo que el buscador estándar de este repositorio ofrece, al analizar no sólo los campos relacionados con el video, sino los contenidos asociados al mismo como los comentarios o los subtítulos. Además, ofrecerá una respuesta resumida, centrándose en aquellos momentos que

realmente son relevantes para la búsqueda, y procesando el análisis de un modo ágil y sencillo.

5.1.1.3 Definiciones, acrónimos y abreviaturas

Durante el presente documento de especificación de requerimientos se trabaja con un conjunto de elementos que a continuación se definen:

- **Definiciones:**

Elemento	Definición
Usuario	Es aquella persona que va a hacer uso del sistema.
Autor	Persona que sube un video a YouTube. No tiene por qué coincidir con la persona que graba el video o que es protagonista.
Equipo de desarrollo	Equipo de desarrollo de los requerimientos y del diseño del proyecto.

Tabla 23: Definiciones del SRS.

- **Acrónimos*:***

Acrónimo	Definición
XML	*Extensible Markup Language* – Lenguaje de marcas extensible
API	*Application Programming Interface* – Interfaz de Programación de Aplicaciones
URL	*Uniform Resource Locator*

Tabla 24: Acrónimos del SRS.

5.1.1.4 Referencias

Título
IEEE Recommended Practice for Software Requirements Secifications
Número de Reporte
IEE 830-1998
Fecha de publicación
Año 1998
Organización
IEEE Standards Association
Fuente
http://standards.ieee.org/findstds/standard/830-1998.html (Última fecha de consulta: 22 de Mayo de 2011)

Tabla 25: Referencia IEEE SRS.

Título
Software engineering -- Product quality -- Part 1: Quality model
Número de Reporte
ISO/IEC 9126-1:2001
Fecha de publicación
15-06-2001

Organización
International Organization for Standarization
Fuente
http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=22749 (Última fecha de consulta: 22 de Mayo de 2011)

Tabla 26: Referencia de Product Quality.

Título
Aplicaciones Informáticas para Personas con Discapacidad – Requisitos de accesibilidad para contenidos en la Web
Número de Reporte
UNE 139803:2004
Fecha de publicación
Diciembre de 2004
Organización
Instituto Nacional de Tecnologías de la Comunicación
Fuente
http://www.inteco.es/Accesibilidad/Normativa_1/Descarga/DescargaUNE_139803 (Última fecha de consulta: 22 de Mayo de 2011)

Tabla 27: Referencia de Requesitos de Accesibilidad.

Título
Esquema de Categorías de YouTube
Número de Reporte
2007
Fecha de publicación
Año 2007
Organización
Google Inc.
Fuente
http://gdata.YouTube.com/schemas/2007/categories.cat (Última fecha de consulta: 22 de Mayo de 2011)

Tabla 28: Referencia de esquema de categorías.

5.1.1.5 Visión General

En las próximas dos secciones de este documento se realiza una descripción del entorno general del proyecto, explicando los factores que influyen en los requerimientos del sistema, como son las funciones del producto o las características de los usuarios del mismo. A continuación, se realiza un esquema de funcionamiento general de la aplicación y se definen de forma precisa el conjunto de requerimientos de los módulos que la componen.

La explicación de las funcionalidades de la aplicación se realiza dividiéndola en módulos, y explicando el propósito, las entradas, las salidas y las funcionalidades de cada uno de ellos.

5.1.2 Descripción general

En esta sección se describen el conjunto de factores que afectan al producto y a sus requerimientos, estableciendo un marco sobre el que desarrollar el detalle de los mismos en apartados posteriores.

5.1.2.1 Perspectiva del producto

El producto que aquí se describe interactuará únicamente con el repositorio de videos online YouTube.

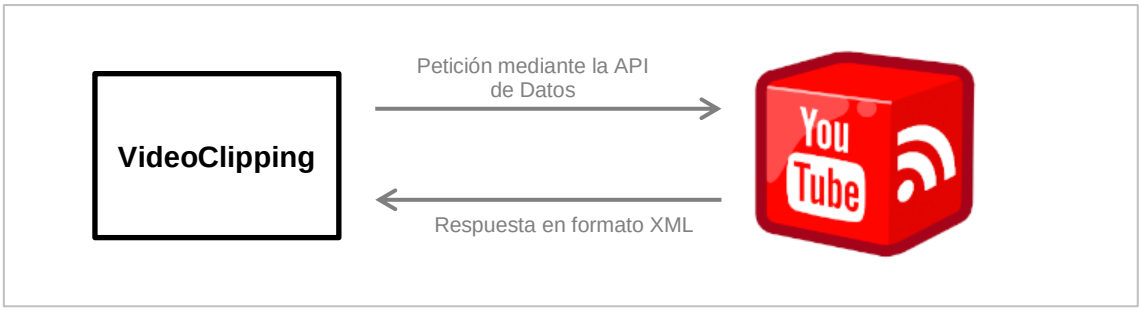


Ilustración 32: Perspectiva del producto.

La interfaz de comunicación con este repositorio se realizará mediante la API desarrollada por el equipo de desarrollo de esta empresa, que recibe el nombre de *API de Datos YouTube*. La petición se habrá de realizar mediante la sintaxis específica definida para esta API, y la respuesta se recibirá en formato XML.

5.1.2.2 Funciones del producto

Dada una configuración de filtrado y unos parámetros de búsqueda iniciales, el sistema realizará consultas de datos a YouTube mediante su API con el objetivo de discriminar el contenido relevante, procesarlo, y obtener un video final con extractos de los videos obtenidos.

Se presenta la ilustración siguiente a modo de esquema general de funcionamiento.

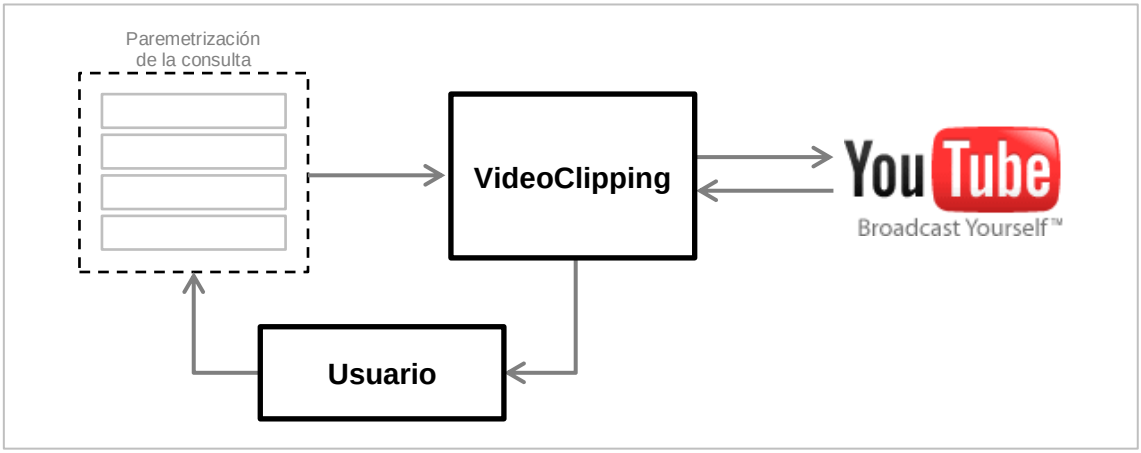


Ilustración 33: Esquema general de funcionamiento.

El usuario establecerá, haciendo uso de la interfaz del sistema VideoClipping, una serie de parámetros de búsqueda de los conceptos de interés. Estos parámetros serán procesados por el sistema, que enviará a YouTube un conjunto de consultas mediante la API de datos.

El servidor de videos retornará los resultados en forma de XML, que el sistema procesará para obtener aquellos fragmentos de video que realmente son relevantes en función de los parámetros de búsqueda y de los contenidos asociados al conjunto de videos recuperado.

Una vez procesada y estructurada la información, se generará un nuevo video que se presentará al usuario, y que estará formado por extractos de los videos recuperados de YouTube.

5.1.2.3 Características del usuario

El sistema VideoClipping se destina a usuarios con un nivel educacional medio, con experiencia en el entorno web, y con conocimiento de la búsqueda de contenido audiovisual en YouTube y en otros repositorios de videos.

El nivel de conocimiento técnico de estos usuarios es medio, pero con interés profesional o a nivel particular en el análisis de los resultados obtenidos mediante esta herramienta.

5.1.2.4 Restricciones

En cuanto a las restricciones del sistema, únicamente es necesario destacar el siguiente aspecto:

- El sistema VideoClipping se ha de desarrollar en un entorno Web de acceso público y gratuito.

5.1.2.5 Suposiciones y Dependencias

- **Suposiciones:**

Se supone que no habrá cambios en el tratamiento de ficheros de subtítulos para los contenidos de YouTube, estando disponible de forma permanente y pública el acceso a estos elementos mediante una URL.

Del mismo modo de asume que los requerimientos presentados en este documento son estables, si bien éste es un proyecto de investigación que está abierto a modificaciones en fases posteriores de la ejecución del proyecto.

- **Dependencias:**

Este sistema dependerá del funcionamiento de la API de Datos de YouTube. Por ello, cambios en la sintaxis de uso o en la ubicación de los objetos relacionados con los videos tienen que ser tenidos en cuenta por el sistema VideoClipping, que ha de modificar su funcionamiento en consecuencia.

5.1.3 Requerimientos Específicos

La definición de los requerimientos específicos se ha realizado teniendo en cuenta el esquema de funcionamiento global del sistema presente en la siguiente página. La explicación de las Entradas, Salidas y Funcionalidades de cada uno de los módulos se describe en los siguientes apartados, dentro la sección de Requisitos Funcionales. Otros Requisitos a tener en cuenta por el sistema quedan contemplados posteriormente en este mismo documento. Todos los requerimientos que a continuación se exponen son de obligado cumplimiento por el sistema.

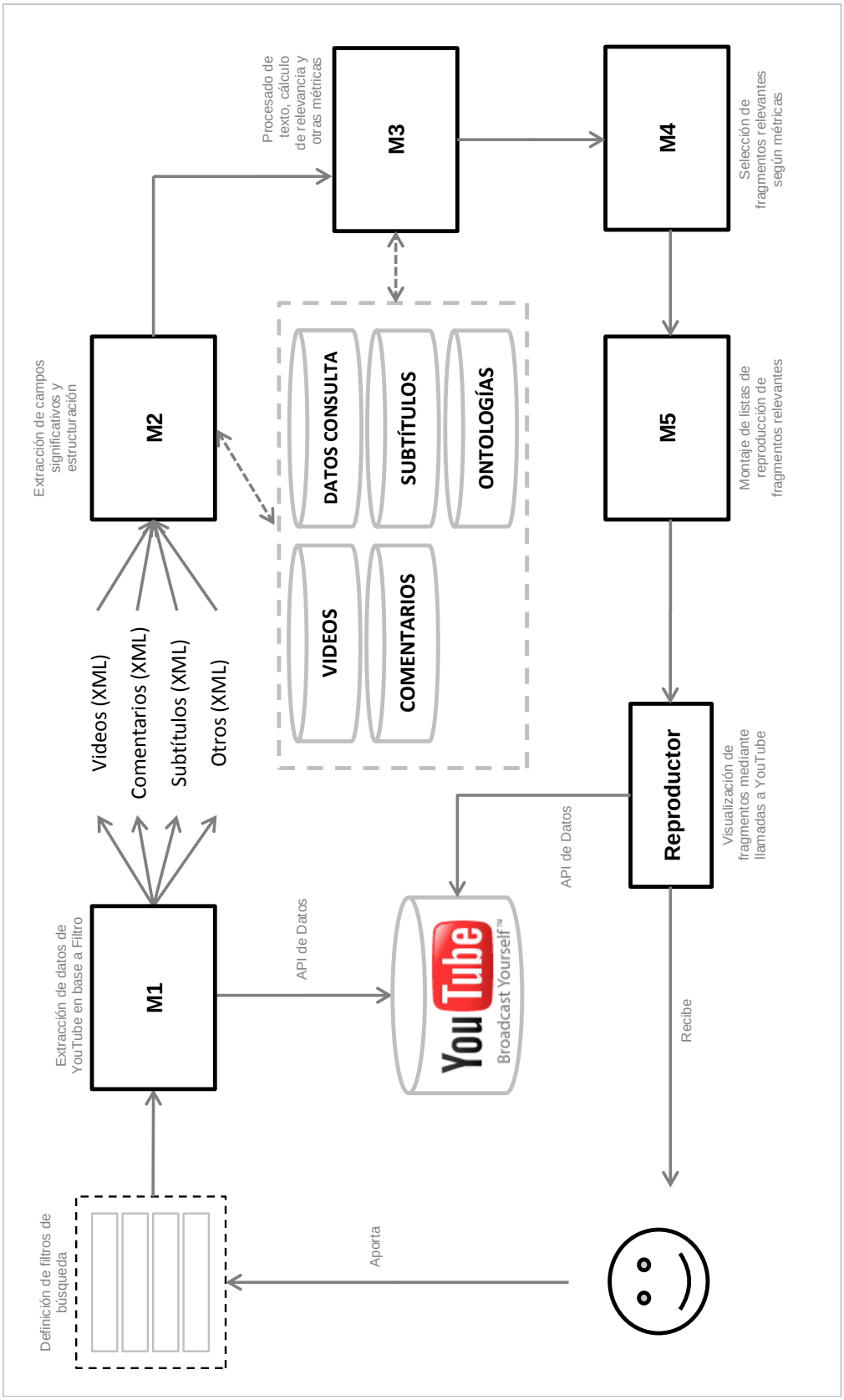


Ilustración 34: Esquema de funcionamiento global.

5.1.3.1 Requisitos Funcionales

- **Modelo M1:**

Este módulo es el encargado de realizar la recuperación de datos de YouTube para almacenar los resultados de forma local.

El conjunto de Entradas, Salidas y Funcionalidades de *M1* queda definido a continuación:

Entradas
(Entrada M1.01): Conjunto de parámetros de filtrado de los campos de YouTube, compuesto por: - **Palabras clave**: Hasta un máximo de 30 caracteres. Palabras claves compuestas separadas por espacios, palabras claves no compuestas separadas por comas. Campo obligatorio sin valores por defecto. - **Categoría**: Un mínimo de una, un máximo de 5. Los valores disponibles serán los que se publican en el esquema de categorías de YouTube (ver Referencias). - **Idioma:** Se define el posible idioma de búsqueda. Este campo únicamente ha de disponer del idioma Español como valor seleccionable, pero se establece para futuras ampliaciones del sistema. Por lo tanto "Español" es el valor por defecto no modificable por el momento. - **Restricción:** Definen 3 niveles de restricción, *none, moderate* o *strict,* directamente aplicables mediante la API de Datos. Por defecto la búsqueda se hace con filtro *moderate*. - **Tiempo:** Filtro en función de si se quieren videos publicados el mismo día, la misma semana, el mismo mes, o en cualquier momento del tiempo. Por defecto se busca en cualquier momento del tiempo. - **Autor:** Campo opcional que permite filtrar por el usuario de YouTube que ha publicado el video. - **Favorables:** Indicar si queremos encontrar resultados Favorables o Desfavorables para el término a buscar, o bien si los queremos ver todos independientemente de los comentarios asociados a los mismos. Se consideran Favorables aquellos videos de los que se desprenden comentarios de usuarios que pueden beneficiar la imagen del concepto a buscar, mientras que Desfavorables son aquellos de los que se extrae una opinión negativa o de burla.
(Entrada M1.02): IP del cliente que realiza la consulta.
(Entrada M1.03): Fecha y hora de la realización de la consulta.
(Entrada M1.04): Tipo de dispositivo con el que se realiza la consulta. Puede ser mediante Navegador o mediante dispositivo móvil.

Salidas
(Salida M1.01): Archivos XML con toda la información que aporte la API de datos de YouTube acerca de los videos consultados teniendo en cuenta los parámetros de filtros definidos.
(Salida M1.02): Archivos XML con los comentarios de cada uno de los videos. Inicialmente habrá un máximo de 4 XMLs por cada video, en bloques de 50 entradas de comentarios por cada uno para poder tener las últimas 200 entradas. En caso de no llegar a este número se guardarán únicamente los XMLs necesarios para contener todos los comentarios disponibles, siempre en bloques de un máximo de 50 comentarios por archivo.
(Salida M1.03): Archivo XML con los subtítulos de cada uno de los videos. Habrá un XML de subtítulos por cada video. En caso de no estar disponible el subtítulo en el idioma seleccionado inicialmente, se realizará una traducción automática de los subtítulos por medio de una consulta a la API de Datos, y se almacenará el subtítulo traducido.
(Salida M1.04): Archivo XML con información relativa a los datos de la consulta: parámetros de los filtros establecidos, IP, fecha y hora y tipo de dispositivo del usuario.

Funcionalidades
(Fun M1.01): Un usuario realizará por medio de un formulario la entrada de los datos del filtro de la consulta. El tipo de datos y la longitud de la consulta están limitados al formato de la entrada[16].
(Fun M1.02): Este módulo realizará la consulta a YouTube por medio de la API de datos, para obtener datos de los últimos 200 videos[17] teniendo en cuenta los filtros establecidos por el usuario, ordenando la consulta por el número de visualizaciones (se retornarán los videos más vistos) y añadiendo además la condición necesaria de recuperar videos con subtítulos.
(Fun M1.03): En función del tipo de acceso del usuario (mediante Navegador o mediante dispositivo móvil) se añadirá a la consulta el parámetro que establezca el requerimiento de formato compatible con el dispositivo de acceso.
(Fun M1.04): Tras la ejecución de la consulta, se guardarán los datos relativos al contenido de videos, comentarios de cada video y archivo de subtítulos de cada video (en el idioma definido en el filtro de entrada) en un repositorio local.
(Fun M1.05): Consultar el primer XML de comentarios recibido para encontrar la ruta a las páginas de los diferentes bloques de comentarios necesarias para obtener los últimos 200 comentarios disponibles de cada video.
(Fun M1.06): Los archivos de subtítulos y de comentarios guardados en formato XML se han de poder relacionar de forma unívoca con el archivo XML de cada uno de los videos.
(Fun M1.07): Se guardarán otros datos de la consulta (parámetros de filtros, IP del cliente, fecha y hora de la consulta y tipo de dispositivo) en otro archivo XML.

[16] Según lo definido en *Entrada M1.01*
[17] Este número puede verse modificado en función del rendimiento del prototipo.

- **Módulo M2:**

Este es el módulo encargado de la extracción de los campos de los archivos XML y su almacenamiento en las bases de datos.

El conjunto de Entradas, Salidas y Funcionalidades de *M2* queda definido a continuación:

Entradas
(Entrada M2.01): XMLs con información relativa a los **Videos** recuperados con la consulta a YouTube del módulo anterior.
(Entrada M2.02): XMLs con información relativa a los **Comentarios** de cada video recuperados con la consulta a YouTube del módulo anterior. Habrá un máximo de 4 XMLs por video, con un máximo de entradas en cada uno de ellos de 50.
(Entrada M2.03): XML con información relativa a los **Subtítulos** de cada video recuperados con la consulta a YouTube del módulo anterior.
(Entrada M2.04): XML con información relativa a la propia consulta realizada por el usuario almacenada en el XML de otra información extraída a través del módulo anterior.

Salidas
(Salida M2.01): Base datos de **Videos** con información estructurada acerca de los diferentes parámetros disponibles de cada uno de ellos.
(Salida M2.02): Base de datos de **Comentarios**, con información del número total de comentarios asociados al video y el contenido de los últimos 200 comentarios relacionados con cada video. De cada comentario almacenado en la base de datos guardar, además del Contenido, la Fecha y el Autor.
(Salida M2.03): Base de datos de **Subtítulos** de los videos. Esta base de datos contendrá, por cada archivo de subtítulos de video: - Listado de frases que lo forman. - Marca de tiempo de inicio de cada frase según consta en el archivo de Subtítulos.
(Salida M2.04): Base de datos de **Datos de Consulta**, con información de los parámetros seleccionados, la IP del usuario, la fecha, la hora de la consulta y el tipo de dispositivo, además de un identificador único de cada consulta.

Funcionalidades
(Fun M2.01): Guardar información relativa a los comentarios, procesando el máximo de 4 archivos de comentarios resultantes del módulo anterior para almacenar datos de un máximo de 200 comentarios.
(Fun M2.02): Preprocesar los sutítulos para identificar las diferentes frases en las entradas del archivo de subtítulos para poderlas almacenar en una base de datos de subtítulos. Guardar además la marca de tiempo de inicio de la frase considerando como inicio el elemento que contiene el inicio de cada frase, por mucho que el comienzo de la misma se encuentre en la mitad del contenido de una entrada de subtítulo.

(Fun M2.03): Estructurar toda la información obtenida mediante el XML de datos del video del módulo anterior en una base de datos de videos.
(Fun M2.04): Almacenar la información asociada a la consulta, contenida en el XML de Otros Datos resultante del módulo anterior.
(Fun M2.05): Los datos de la consulta, de los comentarios, de los subtítulos y de los propios videos almacenados en las tres bases de datos se han de poder relacionar entre sí, pudiendo identificar de una consulta específica los videos recuperados, los subtítulos y los comentarios asociados.

- **Módulo M3:**

Este es el módulo que se encarga del procesado del texto para el cálculo de la relevancia.

Entradas
(Entrada M3.01): Base de datos con información de Videos resultante del módulo anterior.
(Entrada M3.02): Base de datos con información de Comentarios resultante del módulo anterior.
(Entrada M3.03): Base de datos con información de Subtítulos resultante del módulo anterior.
(Entrada M3.04): Base de datos con información de Datos de Consulta resultante del módulo anterior.

Salidas
(Salida M3.01): Base de Datos de Video modificada con información de la relevancia del Video en función de los datos asociados al mismo (ver Fun M3.02) y de los parámetros de búsqueda.
(Salida M3.02): Base de datos de Datos de Consulta modificada con información estadística del tipo de comentarios obtenido (ver Fun M3.04).

Funcionalidades
(Fun M3.01): Procesar los comentarios en busca de estructuras sintácticas que puedan determinar si su contenido es positivo, negativo o neutral.
(Fun M3.02): Medir la relevancia de todos los videos, teniendo en cuenta los siguientes atributos: - Número de veces que el video ha sido visto. - Número de veces que el video ha sido considerado Favorito. - Número de veces que el video ha sido considerado puntuado. - Número de comentarios favorables y desfavorables en relación con el total de comentarios. Teniendo en cuenta lo anterior hay relevancia positiva (Son relevantes positivos aquellos de los que se derivan comentarios positivos hacia la marca o concepto a buscar), y relevancia negativa (Son relevantes negativos aquellos videos de los que se derivan comentarios negativos hacia la marca o concepto).

(Fun M3.03): Guardar el valor de la relevancia obtenida de cada video en la base de datos de Videos, tanto para opiniones favorables, desfavorables y no determinadas.
(Fun M3.04): Guardar en la base de datos de Datos de Consulta el porcentaje de comentarios positivos, negativos y no determinados que se derivan del total de los videos obtenidos en la consulta, con respecto al total de comentarios obtenido.

- **Módulo M4:**

Este módulo se encarga de la selección de los fragmentos relevantes según las métricas obtenidas en los módulos anteriores.

Entradas
(Entrada M4.01): Base de datos de Videos con información de relevancia de cada uno de ellos.
(Entrada M4.02): Base de datos de Datos de Consulta, con información de los parámetros de la consulta.
(Entrada M4.03): Base de datos de Subtítulos, con información separada en frases con líneas de tiempo.
(Entrada M4.04): Base de datos de Ontologías, con la información necesaria para el tratamiento de los sinónimos.

Salidas
(Salida M4.01): Selección de videos con Identificador y Marca de tiempo almacenados en la base de datos Datos de Consulta.
(Salida M4.02): Base de datos de Datos de Consulta modificada con información estadística del tipo de comentarios obtenido (ver Fun M4.04).

Funcionalidades
(Fun M4.01): Buscar sinónimos de los términos de búsqueda almacenados en la base de datos Datos de Consulta, haciendo uso de la Base de Datos de Ontologías.
(Fun M4.02): Obtener el listado de los videos ordenados por relevancia, en función del filtro *Favorables* inicial almacenado en la base de datos de Datos de Consulta.
(Fun M4.03): Lematizar el parámetro de búsqueda y sus sinónimos, y buscarlos en el conjunto de Frases almacenado en la Base de Datos de Subtítulos. Al hacer la búsqueda lematizar también los términos de las frases para buscar las coincidencias.
(Fun M4.04): Guardar las coincidencias en un listado, en el que conste el identificador del video. Además, han de figurar dos marcas de tiempo, la de inicio y la de final.
(Fun M4.05): La marca de tiempo de inicio será la correspondiente a la frase anterior a aquella en que se ha encontrado la coincidencia de lemas. La de final al punto de comienzo de la siguiente frase desde el punto de coincidencia de lemas.

(Fun M4.06): Limitar el listado a un máximo de 50 videos.
(Fun M4.07): Guardar este listado en la base de datos Datos de Consulta.

- **Módulo M5:**

Finalmente, este es el módulo encargado de mostrar el resultado final obtenido tras el proceso de búsqueda y análisis:

Entradas
(Entrada M5.01): Base de datos de Datos de Consulta, con listado ordenado de videos en función de la búsqueda y el rocesado y parámetros de la consulta.
(Entrada M5.02): Base de datos de Datos de Videos, con información de los datos de relevancia de cada video.
(Entrada M5.03): Base de datos de Subtítulos, con información separada en frases con líneas de tiempo.

Salidas
(Salida M5.01): Reproductor de videos relevantes. Concatenación de los fragmentos de videos del listado.
(Salida M5.01): Panel de estadísticas de videos con información de: - Tiempo invertido el procesamiento de la consulta. - Número de videos totales obtenidos. - Métricas de relevancia de cada video reproducido. - Listado de los títulos de los videos reproducidos y enlace al video original. - Muestra de los porcentajes de Comentarios favorables y desfavorables con respecto a los términos de búsqueda.

Funcionalidades
(Fun M5.01): Recuperar el listado de videos y mostrar en un reproductor la concatenación de los fragmentos de video que contiene.
(Fun M5.02): Mostrar información estadística de los videos, tanto a nivel global (datos de toda la búsqueda) como a nivel específico (datos del video). (ver Salida M5.01)

5.1.3.2 Requisitos de Fiabilidad

- **(Req-F.01):** Debido al carácter profesional del usuario del sistema, VideoClipping ha de estar disponible las 24 horas del día, con un máximo de 1 hora semanal de parada por motivos de mantenimiento.

- **(Req-F.02):** No ha de tener restricciones en cuanto al número de usuarios conectados al mismo tiempo.

5.1.3.3 Requisitos de Usabilidad

- **(Reg-U.01):** Teniendo en cuenta las características de los usuarios, el portal ha de ser de fácil manejo, y ha de superar la norma *UNE 139803:2004* de requisitos de accesibilidad para contenidos en la Web.

- **(Reg-U.02):** El sistema ha de poder ser visualizado mediante dispositivos móviles con sistema operativo *Android* i *iOS*, opcionalmente puede estar preparado para otros dispositivos.

5.1.3.4 Requisitos de Eficiencia

- **(Req-E.01):** Como requerimiento inicial se establece que el tiempo de respuesta de las consultas realizadas ha de ser razonable, no superando en ningún caso los 20 segundos. Debido a que este es un proyecto de I+D, este requerimiento puede ser modificado en función del rendimiento del prototipo.

- **(Req-E.02):** Se ha de informar al usuario del tiempo restante estimado para la obtención de los resultados.

5.1.3.5 Requisitos de Mantenimiento

- **(Req-M.01):** Al tener una alta dependencia de la API de Datos de YouTube, será necesario realizar actualizaciones del Sistema para adaptarse a cambios en la versión de dicha API. El sistema ha de poder ser actualizado en este sentido.

5.1.3.6 Requisitos de Portabilidad

- No existen requisitos en este sentido.

5.2 Análisis de Requerimientos y Diseño general

Una vez finalizado el detalle de la especificación de los requisitos del sistema mediante el documento de especificación de requerimientos expuesto en los apartados anteriores, a continuación se realiza un análisis de los requerimientos y un diseño general de la aplicación.

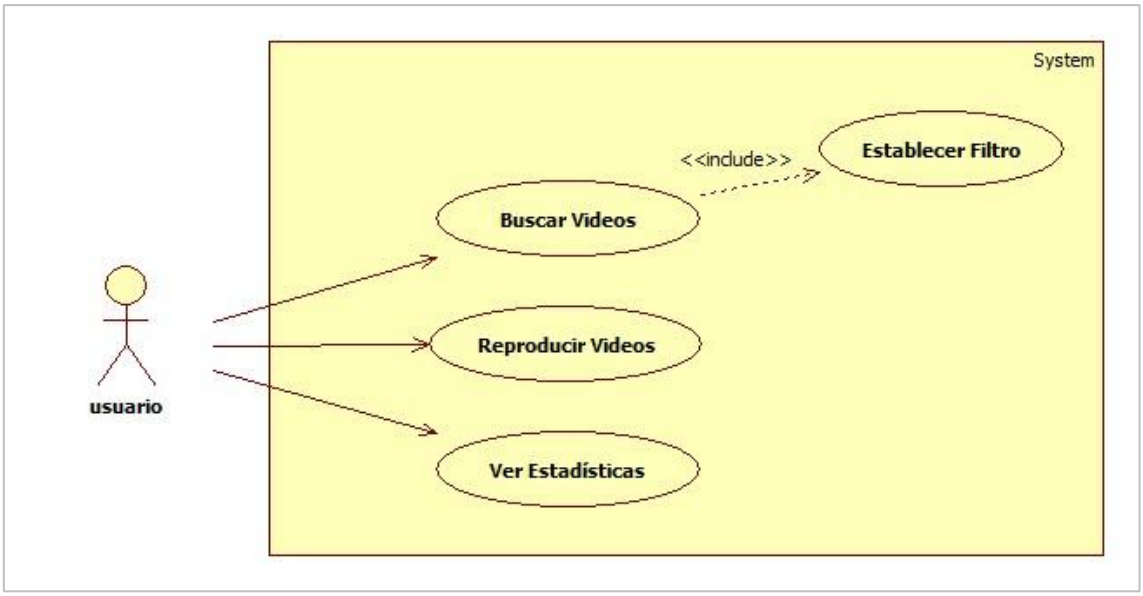
Este análisis se divide diversas secciones. En primer lugar se realiza un análisis de los casos de uso relativos al diseño general de la aplicación. Posteriormente se analiza cada uno de los módulos que forman el sistema, de acuerdo con el esquema general de funcionamiento, identificando los casos de uso de cada uno de ellos y realizando una descripción textual de cada caso de uso.

Conviene tener en cuenta que, en algunos de los diagramas de casos de uso que a continuación se muestran, se define a los usuarios como los propios módulos, al disparar estos elementos la ejecución de las acciones, aun cuando no son estos módulos un usuario como tal.

5.2.1 General

Esta sección engloba el análisis de los requerimientos genéricos del sistema, desde el punto de vista del usuario de la aplicación VideoClipping.

5.2.1.1 Diagrama de Casos de uso



Como se puede observar, el usuario puede realizar tres tipos de acciones: Buscar Videos, Reproducir Videos y Ver Estadísticas de los videos buscados. Para poder realizar una búsqueda de videos es necesario establecer un conjunto de parámetros sobre el que filtrar la información obtenida.

Estas tres acciones, detalladas en el siguiente apartado, se dividen en un conjunto de procesos que se dividen en módulos, que engloban a su vez diferentes actividades a ejecutar. El detalle de estos módulos y sus correspondientes actividades asociadas es también detallado en apartados posteriores.

5.2.1.2 Descripción textual de Casos de Uso

Caso-01: Buscar Videos

Actor Principal: Usuario.

Activación: El usuario selecciona la opción “Buscar Videos” de la aplicación.

Precondición: La aplicación se ha cargado de forma completa y con éxito.

Escenario de éxito principal:

1. El usuario establece un filtro.
2. Se muestra al usuario una pantalla de espera con el tiempo estimado restante de procesado de la consulta.
3. El sistema realiza la búsqueda de los videos mediante la API de datos de

YouTube.

4. El sistema recupera los datos de la respuesta de la API en forma de ficheros XML que guarda en un repositorio local.
5. El sistema guarda en un XML los parámetros de la consulta realizada por el usuario así como otra información del usuario.
6. El sistema procesa los archivos XML almacenados e inserta la información en las bases de datos.
7. El sistema realiza un cálculo del atributo de relevancia de cada video.
8. El sistema recupera un listado los videos ordenados por relevancia según las métricas calculadas y el filtro establecido por el usuario.
9. El sistema Lematiza los términos de la entrada y sus sinónimos y los compara con el contenido de las entradas de los subtítulos para buscar coincidencias.
10. El sistema muestra al usuario un reproductor que reproduce el video con los fragmentos coincidentes obtenidos ordenados por relevancia.
11. El sistema muestra al usuario datos estadísticos de los videos obtenidos.

Extensiones:

*: El usuario cancela la búsqueda en curso.

1. Se cancela la búsqueda y se vuelve a la pantalla principal informando al usuario que la operación ha sido cancelada.

4a: No se puede establecer la conexión con la API de datos de YouTube por un error.

1. El sistema muestra al usuario información acerca del error.
2. Se finaliza el proceso de búsqueda volviendo a la pantalla inicial.

4b: La petición a la API de Datos de YouTube no retorna ningún resultado.

1. El sistema informa al usuario que no se ha podido encontrar ningún resultado.
2. Se finaliza el proceso de búsqueda volviendo a la pantalla inicial.

10a: No se obtienen coincidencias de los términos lematizados y sus sinónimos con respecto a los contenidos de los subtítulos.

1. El sistema informa al usuario que no se han podido encontrar los términos de búsqueda en los videos resultantes.
2. El Sistema finaliza el proceso de búsqueda volviendo a la pantalla inicial.

Postcondición: El usuario obtiene un video con extractos de audiovisuales relevantes para su búsqueda.

Caso-02: Reproducir Videos

Actor Principal: Usuario.

Activación: El usuario selecciona la opción “Reproducir Videos” de la aplicación.

Precondición: La búsqueda realizada ha obtenido resultados satisfactorios.

Escenario de éxito principal:

1. El sistema reproduce los extractos de los videos relevantes obtenidos.

Extensiones:

*: El usuario pausa la reproducción en curso.

1. El Sistema para la reproducción en curso sin perder el instante de reproducción actual ni cambiar de fragmento.

Postcondición: -.

Caso-03: Ver estadísticas

Actor Principal: Usuario.

Activación: El usuario selecciona la opción de “Ver estadísticas” de la aplicación.

Precondición: La búsqueda realizada ha obtenido resultados satisfactorios.

Escenario de éxito principal:

1. El sistema muestra datos estadísticos del total de videos obtenidos.
2. El sistema muestra un listado de los videos de origen de los fragmentos obtenidos junto con un enlace a la dirección del video original en YouTube.

Postcondición: -.

Caso-04: Establecer Filtro

Actor Principal: Usuario.

Activación: El usuario completa los datos de búsqueda en el formulario.

Precondición: La aplicación se ha cargado de forma completa y con éxito.

Escenario de éxito principal:

1. El usuario establece un valor para los diferentes campos del filtro.
2. El sistema verifica la inserción de datos en los campos obligatorios.
3. El sistema verifica la corrección del tipo de dato de los campos obligatorios y opcionales.

Extensiones:
2a: Alguno de los campos obligatorios no contienen datos.
1. Se muestra un mensaje junto a cada campo obligatorio sin datos informando al usuario de su obligatoriedad.

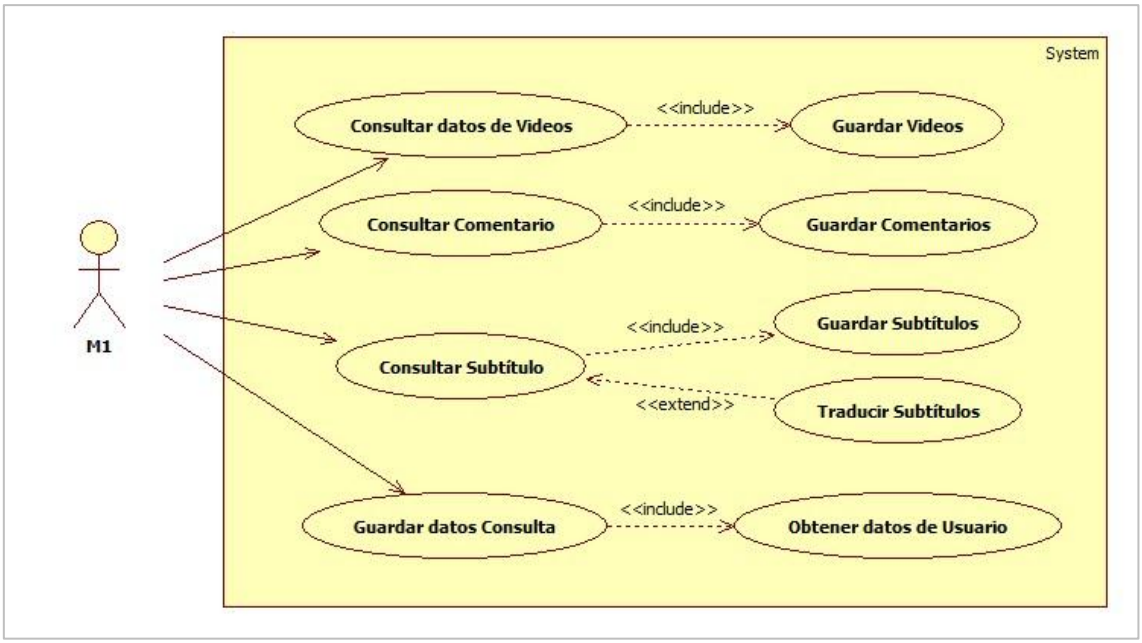
3a: Alguno de los tipos de los valores introducidos no cumple con el tipo de valor requerido para el campo.
1. El Sistema muestra un mensaje junto a cada campo con error en el tipo de valor introducido informando al usuario del error.

Postcondición: El filtro establecido por el usuario contiene valor en los campos obligatorios y el tipo de valor en todos los campos introducidos es válido.

5.2.2 Módulo M1

El módulo M1 se encarga de la consulta de los datos a YouTube mediante la API de Datos y del almacenamiento de los resultados obtenidos.

5.2.2.1 Diagrama de casos de uso



Respecto al funcionamiento de este módulo conviene tener en cuenta que en algunos casos la recuperación de datos no es directa, sino que es necesario realizar operaciones intermedias.

Este es el caso de los comentarios, que hay que consultar mediante datos recuperados en la consulta del video, o de los subtítulos, que es necesario traducir en algunas ocasiones.

5.2.2.2 Descripción textual de Casos de Uso

M1-Caso01: Consultar datos de Videos

Actor Principal: Módulo M1.

Activación: Módulo M1 ejecuta un proceso de búsqueda.

Precondición: Se ha establecido un filtro de videos válido conforme a los requerimientos de la API de Datos de YouTube.

Escenario de éxito principal:

1. El Sistema realiza una consulta de videos a la API de Datos de YouTube teniendo en cuenta los datos del filtro establecidos.
2. La API de Datos retorna un conjunto de valores correspondiente a resultado

de la consulta de videos.
3. El Sistema guarda los resultados.

Extensiones:
2a: No se obtienen datos como resultado de la consulta.
1. El sistema muestra un mensaje al usuario informando de que no se han encontrado entradas de videos para el filtro establecido.
2. El Sistema finaliza el proceso de búsqueda volviendo a la pantalla inicial.

Postcondición: Se ha ejecutado una consulta válida y los resultados de la misma han sido almacenados satisfactoriamente.

M1-Caso02: Guardar Videos

Actor Principal: Módulo M1.

Activación: Tras la ejecución de la consulta de datos de videos derivada de la ejecución del proceso de búsqueda.

Precondición: Se ha obtenido un conjunto no vacío de datos correspondiente al resultado de la consulta de videos.

Escenario de éxito principal:

1. El sistema recupera los datos de la consulta.
2. El sistema almacena los datos en un repositorio local en el mismo formato XML en que son recuperados.

Extensiones: -

Postcondición: Se ha guardado un fichero que contiene los datos resultantes de la consulta de videos en formato XML.

M1-Caso03: Consultar Comentarios

Actor Principal: Módulo M1.

Activación: Módulo M1 ejecuta un proceso de búsqueda.

Precondición: Se ha establecido un filtro de videos válido conforme a los requerimientos de la API de Datos de YouTube y se ha almacenado un fichero XML con el resultado de los videos obtenidos.

Escenario de éxito principal:

1. El Sistema consulta el fichero XML de videos almacenado para obtener la lista de identificadores de los videos recuperados.
2. Para cada elemento de esta lista, el Sistema recupera el identificador del video y se realizan consultas a la API de datos mediante este valor para obtener un total de 200 comentarios de cada video.

3. El sistema guarda los resultados.

Extensiones:
2a: No se obtienen 200 comentarios de alguno de los videos.
1. El sistema almacena los comentarios disponibles para el video, aunque no lleguen a las 200 entradas.

2b: No se obtiene ningún comentario para alguno de los videos.
1. El sistema descarta el video, prosiguiendo con el resto de elementos del listado de videos recuperado de la consulta.

Postcondición: Se ha obtenido un listado de los comentarios disponibles para cada uno de los videos resultante de la consulta, y se ha almacenado el listado de forma satisfactoria.

M1-Caso04: Guardar Comentarios

Actor Principal: Módulo M1.

Activación: Tras la ejecución de la consulta de comentarios de videos derivada de la ejecución del proceso de búsqueda.

Precondición: Se ha obtenido un conjunto no vacío de datos correspondiente al resultado de la consulta de comentarios disponibles de cada video.

Escenario de éxito principal:

1. El sistema recupera los datos de la consulta.
2. El sistema almacena los datos en un repositorio local en el mismo formato XML en que los comentarios son recuperados.

Extensiones: -

Postcondición: Se ha guardado un fichero que contiene los datos resultantes de la consulta de comentarios disponibles para cada video en formato XML.

M1-Caso05: Consultar Subtítulos

Actor Principal: Módulo M1.

Activación: Módulo M1 ejecuta un proceso de búsqueda.

Precondición: Se ha establecido un filtro de videos válido conforme a los requerimientos de la API de Datos de YouTube y se ha almacenado un fichero XML con el resultado de los videos obtenidos.

Escenario de éxito principal:

1. El Sistema consulta el fichero XML de videos almacenado para obtener la lista de identificadores de los videos recuperados.
2. Para cada elemento de esta lista, el Sistema recupera el identificador del video

y se realizan consultas a la API de datos mediante este valor para obtener el fichero de subtítulos disponible para el video en el idioma establecido en el proceso de búsqueda.

3. El Sistema guarda los resultados.

Extensiones:
2a: El video no dispone de fichero de subtítulos.
1. El sistema descarta el video, prosiguiendo con la búsqueda de subtítulos a través del resto de elementos del listado de videos recuperado de la consulta.
2. Se elimina el video de la base de datos de Datos de Videos.

2b: El video no dispone del fichero de subtítulos en el idioma de la consulta, pero sí en otro idioma.
1. El sistema realiza una traducción del subtítulo de video al idioma establecido en el filtro de la consulta.
2. El sistema almacena el fichero resultante de la traducción y prosigue con el análisis del resto de videos.

Postcondición: Se ha obtenido un archivo de subtítulos disponible para cada uno de los videos resultantes de la consulta con algún archivo de subtítulos asociado, tanto en el idioma establecido en el filtro de búsqueda como en otro, y se han almacenado los ficheros resultantes de forma satisfactoria.

M1-Caso06: Traducir Subtítulos

Actor Principal: Módulo M1.

Activación: Se analiza un video que dispone de un fichero de subtítulos disponible en un idioma que difiere del idioma establecido en los filtros de búsqueda.

Precondición: Se ha obtenido un identificador de video con un fichero de subtítulos disponible.

Escenario de éxito principal:

1. El Sistema recuperar el fichero de subtítulos disponible.
2. El Sistema analiza los parámetros del fichero para identificar el idioma del mismo.
3. El Sistema realiza una consulta mediante la API de Datos para obtener una traducción automática del fichero de subtítulos del idioma original al idioma definido en los parámetros del filtro de búsqueda.

Extensiones:
2a: No se puede determinar el idioma del fichero de subtítulos.
1. El sistema descarta el video, prosiguiendo con la búsqueda de subtítulos a través del resto de elementos del listado de videos recuperado de la consulta.

3a: La traducción automática del fichero de subtítulos no se puede realizar por no estar disponible la traducción desde el idioma seleccionado o por un error en la consulta.
1. El sistema descarta el video, prosiguiendo con la búsqueda de subtítulos a través del resto de elementos del listado de videos recuperado de la consulta.

Postcondición: Se ha traducido el fichero de subtítulos al idioma definido en los parámetros del filtro de búsqueda.

M1-Caso07: Guardar Subtítulos

Actor Principal: Módulo M1.

Activación: Tras la ejecución de la consulta de subtítulos de videos derivada de la ejecución del proceso de búsqueda.

Precondición: Se ha obtenido un conjunto no vacío de datos correspondiente al resultado de la consulta de subtítulos disponibles de cada video.

Escenario de éxito principal:

1. El sistema recupera los datos de la consulta de subtítulos.
2. El sistema almacena los datos en un repositorio local en el mismo formato XML en que los subtítulos son recuperados.

Extensiones: -

Postcondición: Se ha guardado un fichero que contiene los datos resultantes de la consulta de subtítulos disponibles para cada video, con subtítulos disponibles en algún idioma, en formato XML.

M1-Caso08: Guardar datos Consulta

Actor Principal: Módulo M1.

Activación: Módulo M1 ejecuta un proceso de búsqueda.

Precondición: Se ha establecido un filtro de videos válido conforme a los requerimientos de la API de Datos de YouTube.

Escenario de éxito principal:

1. El sistema recupera los datos del filtro establecido por el usuario.
2. El sistema recupera los datos del usuario.
3. El sistema almacena los datos recuperados en un repositorio local en formato XML.

Extensiones: -

Postcondición: Se ha guardado un fichero que contiene datos de la consulta relativos a los parámetros de búsqueda establecidos y a datos del usuario.

M1-Caso09: Obtener datos de Usuario

Actor Principal: Módulo M1

Activación: Módulo M1 ejecuta un proceso de búsqueda que deriva en un

almacenamiento de datos de la consulta.

Precondición: El entorno en el que se ejecuta la aplicación permite recuperar datos del usuario.

Escenario de éxito principal:

1. El sistema recupera datos del usuario que realiza la consulta a partir de las variables del entorno de ejecución de la aplicación.
2. El sistema establece la localización del usuario que realiza la consulta mediante las variables obtenidas.
3. El sistema almacena los datos recuperados.

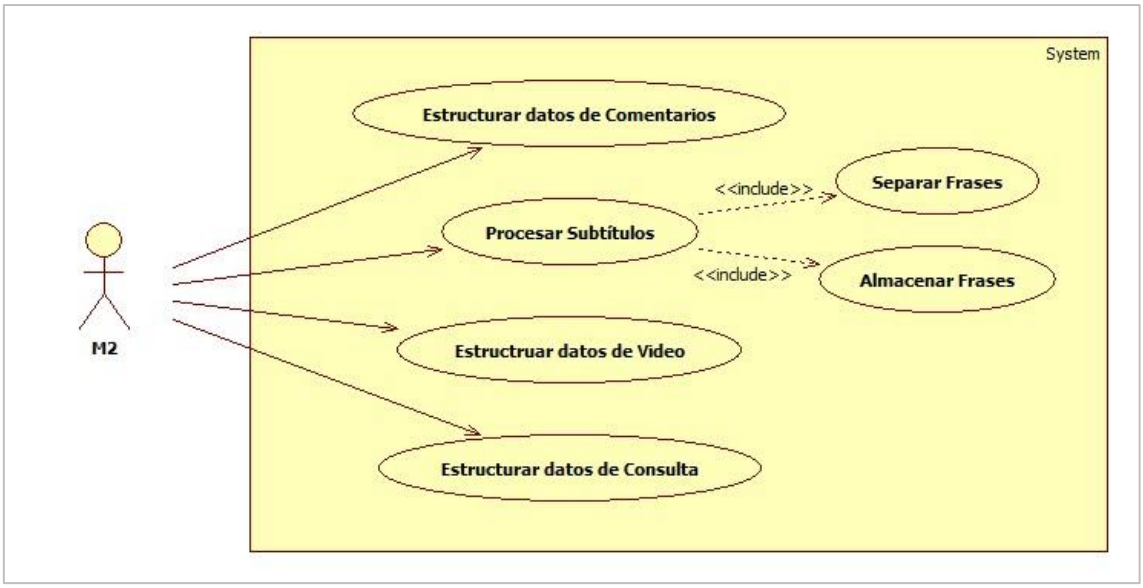
Extensiones:
2a: No se puede establecer la localización a partir de los datos obtenidos.
1. El Sistema almacena un valor de “Indefinido” como atributo de localización.

Postcondición: Se han obtenido datos del usuario que realiza la consulta a partir del entorno de ejecución de la aplicación.

5.2.3 Módulo M2

El módulo M2 se encara de la estructuración de los datos recuperados de la base de datos a partir del módulo M1. Esta estructuración se ve traducida en una inserción de los datos en distintas bases de datos que contienen toda la información recuperada.

5.2.3.1 Diagrama de casos de uso



Tal y como se observa en el diagrama, este mismo módulo realiza un procesamiento de los subtítulos, separando las frases del fichero de subtítulos a través del análisis sintáctico de su contenido. La información almacenada en este caso en la

base de datos contiene información no sólo del contenido en forma de texto, sino de las marcas de tiempo de inicio y de fin de cada una de ellas.

5.2.3.2 Descripción textual de Casos de Uso

M2-Caso01: Estructurar Videos

Actor Principal: Módulo M2

Activación: Módulo M2 realiza una estructuración de los resultados tras ejecución del módulo M1.

Precondición: Se dispone un conjunto de archivos con información de videos almacenados en formato XML.

Escenario de éxito principal:

1. El Sistema *parsea* el archivo XML de cada video.
2. El Sistema consulta los diferentes campos del fichero y almacena los valores en una base de datos de Datos de Videos.

Extensiones: -

Postcondición: Se dispone de una base de datos de Datos de Videos con información relativa al conjunto de videos recuperados tras la consulta.

M2-Caso02: Estructurar datos de Comentarios

Actor Principal: Módulo M2

Activación: Módulo M2 realiza una estructuración de los resultados tras ejecución del módulo M1.

Precondición: Se dispone un conjunto de archivos con información de los comentarios de cada video almacenados en formato XML.

Escenario de éxito principal:

1. El Sistema *parsea* los archivos XML de comentarios cada video.
2. El Sistema consulta los diferentes campos del fichero de comentarios y almacena el autor del comentario y el texto del mismo en una base de datos de Datos de Comentarios.

Extensiones: -

Postcondición: Se dispone de una base de datos de Datos de Comentarios con información relativa al conjunto de videos recuperados tras la consulta.

M2-Caso03: Procesar Subtítulos

Actor Principal: Módulo M2

Activación: Módulo M2 realiza una estructuración de los resultados tras ejecución del módulo M1.

Precondición: Se dispone de un fichero de subtítulos para cada uno de los videos recuperados en la consulta.

Escenario de éxito principal:

1. El Sistema *parsea* el archivo XML de subtítulos cada video.
2. El Sistema procesa cada entrada del fichero de subtítulos separando las frases que componen el texto.
3. El Sistema almacena la información de cada frase en una base de datos.

Extensiones: -

Postcondición: Se dispone de una base de datos de Datos de Subtítulos con información relativa al conjunto de frases que componen los subtítulos de cada video.

M2-Caso04: Separar Frases

Actor Principal: Módulo M2

Activación: Módulo M2 procesa los subtítulos en la estructuración de los resultados.

Precondición: Se dispone de un fichero de subtítulos para cada uno de los videos recuperados en la consulta.

Escenario de éxito principal:

1. El Sistema obtiene el texto del archivo de subtítulos.
2. El Sistema realiza una búsqueda de las estructuras sintácticas para analizar el inicio y el fin de cada frase.
3. El sistema delimita el inicio de cada frase con una marca de tiempo.

Extensiones:
2a: No se puede delimitar el inicio y final de ninguna frase.
1. El sistema descarta el video estableciendo una propiedad de “Video descartado” en la base de datos de Datos de Videos de acuerdo con el identificador del video correspondiente.

Postcondición: Se obtiene un listado de las diferentes frases que componen el fichero de subtítulos, así como una marca de tiempo del inicio de cada una de ellas.

M2-Caso05: Almacenar Frases

Actor Principal: Módulo M2

Activación: Módulo M2 procesa los subtítulos en la estructuración de los resultados.

Precondición: Se dispone, ara cada video, de un listado de las diferentes frases que componen el fichero de subtítulos junto con una marca de tiempo de inicio de cada una de ellas.

Escenario de éxito principal:

1. El consulta las diferentes frases del listado de frases del listado de subtítulos de los diferentes videos.
2. El sistema almacena el contenido de la frase y la marca de tiempo de inicio en la base de datos de Datos de Subtítulos.

Extensiones: -

Postcondición: Se obtiene un listado de las diferentes frases que componen el fichero de sibtítulos, así como una marca de tiempo del inicio de cada una de ellas.

M2-Caso06: Estructurar datos de Consulta

Actor Principal: Módulo M2

Activación: Módulo M2 realiza una estructuración de los resultados tras ejecución del módulo M1.

Precondición: Se dispone de un archivo de datos con información relativa la consulta realizada por el usuario.

Escenario de éxito principal:

1. El Sistema *parsea* el archivo XML correspondiente a la consulta realizada.
2. El Sistema consulta los diferentes campos del fichero y almacena los parámetros del filtro de búsqueda y los datos del usuario en una base de datos de Datos de Consulta.

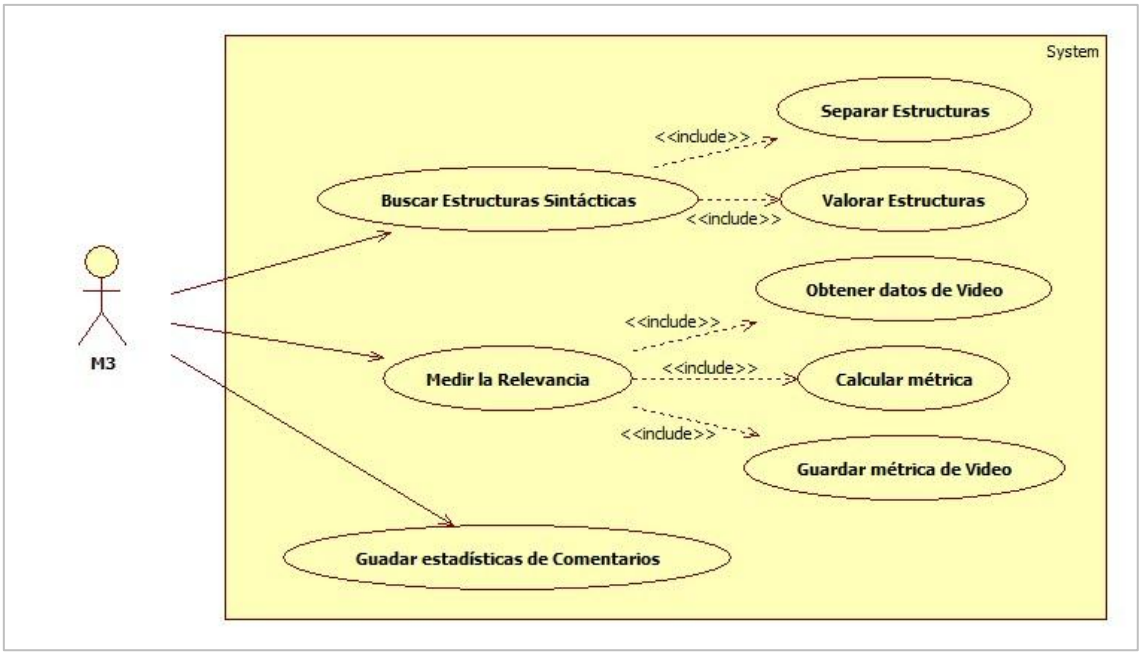
Extensiones: -

Postcondición: Se dispone de una base de datos de Datos de Consulta con información relativa al usuario y a los parámetros de filtrado establecidos en el proceso de búsqueda.

5.2.4 Módulo M3

El módulo M3 realiza un preprocesado de la información contenida en los comentarios para obtener un conjunto de valores que permitan determinar la importancia de un video con respecto a la consulta realizada.

5.2.4.1 Diagrama de casos de uso



En este módulo se procesan las estructuras sintácticas de los comentarios para evaluar cómo de favorables son con respecto al término buscado. Por otro lado se evalúa el valor de Relevancia, que combinando datos estadísticos del video con el valor calculado previamente en los comentarios sirve de grado de importancia del video.

5.2.4.2 Descripción textual de Casos de Uso

M3-Caso01: Buscar Estructuras Sintácticas

Actor Principal: Módulo M3

Activación: Módulo M3 realiza un procesado de los comentarios.

Precondición: Se dispone de una base de datos de Datos de Comentarios, con información relativa al contenido de cada comentario.

Escenario de éxito principal:

1. El Sistema consulta la base de Datos de Comentarios para obtener el contenido de cada comentario de los videos.
2. Para cada comentario, el Sistema realiza una búsqueda de estructuras

sintácticas que puedan determinar si el comentario es favorable o desfavorable.

3. El Sistema realiza una valoración de cada estructura que almacena en la base de datos de Datos de Cometarios.

Extensiones: -

Postcondición: Se dispone de una base de datos de Datos de Comentarios con información relativa a los comentarios asociados a cada video, con información relativa a la valoración favorable o desfavorable de cada comentario.

M3-Caso02: Separar Estructuras

Actor Principal: Módulo M3

Activación: Módulo M3 realiza una búsqueda de estructuras sintácticas.

Precondición: Se dispone de una base de datos de Datos de Comentarios, con información relativa al contenido de cada comentario.

Escenario de éxito principal:

1. El sistema procesa el contenido de cada comentario y trata de encontrar coincidencias con un conjunto de estructuras sintácticas definidas.
2. El sistema considera cada coincidencia con una estructura como un elemento independiente, por lo que almacena cada una de las estructuras en la base de datos de Datos de Comentarios.

Extensiones:
1a: El Sistema no puede determinar ninguna estructura sintáctica.

1. El Sistema descarta el comentario marcando el valor que indica si es favorable o no favorable como “No definido”.

Postcondición: Se dispone de una base de datos de Datos de Comentarios con un listado de las diferentes estructuras sintácticas asociadas a cada comentario de cada video.

M3-Caso03: Valorar Estructuras

Actor Principal: Módulo M3

Activación: Módulo M3 realiza una búsqueda de estructuras sintácticas.

Precondición: Se dispone de una base de datos de Datos de Comentarios que contiene un listado de las diferentes estructuras sintácticas asociadas a cada comentario.

Escenario de éxito principal:

1. El Sistema procesa cada estructura sintáctica de cada uno de los comentarios.

2. El Sistema valora si la estructura sintáctica es favorable o desfavorable.
3. El Sistema almacena el resultado obtenido en la Base de Datos de comentarios, realizando un cómputo del valor (favorable y desfavorable) de las diferentes estructuras sintácticas extraídas de cada comentario.

Extensiones:
2a: El Sistema no puede determinar si la estructura sintáctica es favorable o desfavorable.
1. Se descarta la estructura sintáctica.

3a: El Sistema no puede valorar ninguna estructura sintáctica para el comentario.
1. El Sistema descarta el comentario marcando el valor que indica si es favorable o no favorable como “No definido”.

Postcondición: Se dispone de una base de datos de Datos de Comentarios con un listado de las diferentes estructuras sintácticas asociadas a cada comentario de cada video.

M3-Caso04: Medir la Relevancia

Actor Principal: Módulo M3

Activación: Módulo M3 realiza una medida de la relevancia de los comentarios.

Precondición: Se dispone de una base de datos de Datos de Comentarios que contiene un listado de los comentarios de los diferentes videos, junto con un valor que indica en qué medida el comentario se puede considerar favorable, desfavorable o no definido.

Escenario de éxito principal:

1. El Sistema consulta los datos del Video.
2. El Sistema calcula la métrica de relevancia en función de los datos obtenidos.
3. El Sistema almacena el factor de Relevancia en la Base de Datos de Videos.

Extensiones: -

Postcondición: Se dispone de una base de datos de Datos de Videos con un valor de relevancia asociado a cada uno de ellos.

M3-Caso05: Obtener datos Video

Actor Principal: Módulo M3

Activación: Módulo M3 realiza una medida de la relevancia de los comentarios.

Precondición: Se dispone de una base de datos de Datos de Videos, que contiene información relativa a los videos obtenidos en la consulta.

Escenario de éxito principal:

1. El Sistema consulta los datos estadísticos de acceso y valoración del Video de la base de datos de Datos de Video.
2. El sistema almacena estos resultados para un cálculo de métrica posterior.

Extensiones: -

Postcondición: Se dispone de un listado de los diferentes parámetros de acceso y valoración del video.

M3-Caso06: Calcular métrica

Actor Principal: Módulo M3

Activación: Módulo M3 realiza una medida de la relevancia de los comentarios.

Precondición: Se dispone de un listado de los datos de acceso y valoración del video además de una base de datos de Datos de Comentarios con información relativa a si el contenido de cada comentario es favorable, desfavorable o no definido.

Escenario de éxito principal:

1. El Sistema consulta los datos estadísticos de acceso y valoración del Video así como un cómputo de los valores de cada comentario que determinan si el contenido es favorable, desfavorable o no definido.
2. El sistema realiza una serie de cálculos con estos datos para obtener un valor de relevancia positiva y un valor de relevancia negativa para cada video.

Extensiones: -

Postcondición: Se dispone de un valor de relevancia positiva y de un valor de relevancia negativa para cada video.

M3-Caso07: Guardar métrica de Video

Actor Principal: Módulo M3

Activación: Módulo M3 realiza una medida de la relevancia de los comentarios.

Precondición: Se dispone de un valor de relevancia negativa y de un valor de relevancia positiva para cada video.

Escenario de éxito principal:

1. El Sistema consulta los datos de relevancia de cada video.
2. El Sistema guarda los datos de relevancia en la base de datos de Datos de Video.

Extensiones: -

Postcondición: Se dispone de una base de datos de Datos de Videos con un valor de relevancia asociado a cada uno de ellos.

M3-Caso08: Guardar estadísticas de Comentarios

Actor Principal: Módulo M3

Activación: Módulo M3 guarda estadísticas de los comentarios.

Precondición: Se dispone de una base de datos de Datos de Comentarios con registros de comentarios asociados a cada video.

Escenario de éxito principal:

1. El Sistema consulta los datos de los comentarios de cada video.
2. El Sistema guarda un sumario de los datos de los comentarios asociados a cada video en la base de datos de Datos de Videos.

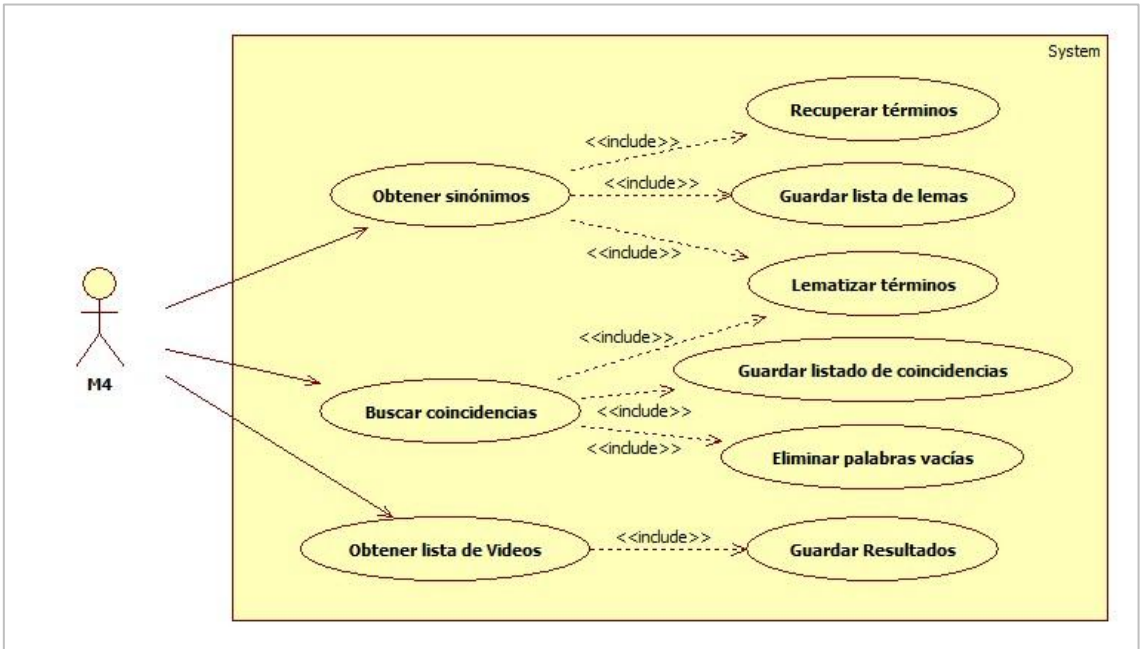
Extensiones: -

Postcondición: Se dispone de una base de datos de Datos de Videos con valores estadísticos relativos al volumen de comentarios asociados a cada video.

5.2.5 Módulo M4

El módulo M4 se encarga de realizar el proceso de búsqueda de los términos de la consulta dentro del texto de los subtítulos de los videos.

5.2.5.1 Diagrama de casos de uso



Para realizar este proceso, el Sistema busca no sólo las palabras establecidas en el filtro inicial, sino también los sinónimos de los términos de búsqueda. Este conjunto de palabras es lematizado y comparado con la lematización del texto de los subtítulos después de haber eliminado las palabras vacías de su contenido.

Finalmente, de todo el proceso se obtiene una lista de videos con coincidencias, que es almacenada en la base de datos de Datos de Consulta para su posterior reproducción en el reproductor de videos.

5.2.5.2 Descripción textual de Casos de Uso

M4-Caso01: Obtener sinónimos

Actor Principal: Módulo M4

Activación: Módulo M4 obtiene sinónimos de los términos de búsqueda.

Precondición: Se dispone de una base de datos de Datos de Consulta con información relativa a los parámetros de búsqueda establecidos.

Escenario de éxito principal:

1. El Sistema consulta los términos por los que el usuario ha filtrado la búsqueda.
2. El Sistema realiza una lematización de los términos de búsqueda.
3. El Sistema realiza una lematización de los sinónimos encontrados.
4. El Sistema obtiene una lista de lemas que contiene datos tanto de los términos de búsqueda como de sus sinónimos.
5. El Sistema almacena la lista de lemas en la base de datos de Datos de Consulta.

Extensiones:
2a: No se obtienen sinónimos para los términos buscados.
1. El Sistema guardaa en la base de datos únicamente los lemas de los términos de búsqueda.

Postcondición: Se dispone de una base de datos de Datos de Consulta con valores relativos a los lemas tanto de los términos de búsqueda como de sus sinónimos.

M4-Caso02: Recuperar términos

Actor Principal: Módulo M4

Activación: Módulo M4 obtiene sinónimos de los términos de búsqueda.

Precondición: Se dispone de una base de datos de Datos de Consulta con información relativa a los parámetros de búsqueda establecidos.

Escenario de éxito principal:

1. El Sistema consulta los términos por los que el usuario ha filtrado la búsqueda.

2. El Sistema obtiene una lista de sinónimos de estos términos consultando la base de datos de Ontologías.
3. El Sistema realiza una lista conjunta de los términos de la búsqueda junto con los términos resultantes de la consulta de sinónimos.

Extensiones:
2a: No se obtienen sinónimos para los términos buscados.
1. El Sistema almacenaa en la lista resultante únicamente los términos de la búsqueda.

Postcondición: Se dispone de lista de términos, que contiene tanto términos de búsqueda como sinónimos.

M4-Caso03: Lematizar términos

Actor Principal: Módulo M4.

Activación: Módulo M4 obtiene sinónimos de los términos de búsqueda o busca coincidencias.

Precondición: Se dispone de lista de términos a lematizar.

Escenario de éxito principal:

1. Por cada término el sistema aplica técnicas de lematización para obtener el *lema* de cada palabra.
2. El sistema almacena la lista de lemas resultantes.

Extensiones:
1a: No se puede lematizar el término.
1. El Sistema almacena en la lista el término original.

Postcondición: Se dispone de una lista de lemas resultante del proceso de lematización.

M4-Caso04: Guardar lista de lemas

Actor Principal: Módulo M4.

Activación: Módulo M4 obtiene sinónimos de los términos de búsqueda.

Precondición: Se dispone de lista de términos lematizados.

Escenario de éxito principal:

1. El Sistema almacena cada término lematizado en la base de datos de Datos de Consulta.

Extensiones: -

Postcondición: Se dispone de una base de datos de Datos de Consulta con una lista de lemas de los términos de búsqueda y sus sinónimos.

M4-Caso05: Obtener lista de Videos

Actor Principal: Módulo M4.

Activación: Módulo M4 obtiene una lista de videos ordenados.

Precondición: Se dispone de una base de datos de Datos de Videos con datos de Videos y de una base de datos de Datos de Consulta con información relativa a la consulta.

Escenario de éxito principal:

1. El Sistema obtiene de la base de datos una lista de los videos ordenados por el valor de relevancia en función de los parámetros de filtrado.
2. El Sistema almacena el listado resultante.

Extensiones: -

Postcondición: Se dispone de una base de datos de Datos de Consulta con una lista de videos ordenados por relevancia en función del filtro de búsqueda inicial.

M4-Caso06: Guardar Resultados

Actor Principal: Módulo M4.

Activación: Módulo M4 obtiene una lista de videos ordenados.

Precondición: Se dispone de una base de datos de Datos de Consulta y se dispone de un listado de videos ordenados por relevancia.

Escenario de éxito principal:

1. El Sistema almacena el listado de videos ordenados por relevancia en la base de datos de Datos de Consulta.

Extensiones: -

Postcondición: Se dispone de una base de datos de Datos de Consulta con una lista de videos ordenados por relevancia en función del filtro de búsqueda inicial.

M4-Caso07: Buscar coincidencias

Actor Principal: Módulo M4.

Activación: Módulo M4 busca coincidencias de lemas.

Precondición: Se dispone de una base de datos de Datos de Consulta con un conjunto de lemas de los términos de entrada y sus sinónimos, además de un listado de videos ordenados por relevancia. Se dispone de una base de datos de Datos de Subtítulos con un conjunto de frases por cada video y una marca de tiempo.

Escenario de éxito principal:

1. El Sistema recupera las frases que componen los subtítulos del video.
2. El Sistema elimina las palabras vacías de las frases.
3. El Sistema compara el listado de términos de obtenido de eliminar las palabras vacías con el listado de términos de la base de datos de Datos de Consulta.
4. En caso de existir coincicencias, el sistema almacena los datos del video coincidente y la marca de tiempo de la frase en un listado.
5. El sistema almacena la marca de tiempo de inicio de la tercera frase posterior a la frase coincidente, y establece esta marca de tiempo como la marca de final del fragmento coincidente.

Extensiones:
4a: No existe ninguna coincidencia de lemas entre los términos buscados y sus sinónimos y los términos contenidos en la frase.
1. El video se descarta, analizando el siguiente video de la lista de videos ordenados por relevancia de la base de datos de Datos de Consulta.

5a: No existen tres frases más de subtítulos.
1. El sistema almacena como marca de tiempo de final de fragmento aquella más lejana posible de las frases disponibles tras la marca de tiempo de la frase coincidente.

Postcondición: Se dispone de un listado de videos con coincidencias entre los lemas de los términos de búsqueda o sus sinónimos y los lemas de los términos de los subtítulos.

M4-Caso08: Guardar listado de coincidencias

Actor Principal: Módulo M4.

Activación: Módulo M4 busca coincidencias de lemas.

Precondición: Se dispone de un listado de videos con coincidencias entre los lemas de los términos de búsqueda o sus sinónimos y los lemas de los términos de los subtítulos. Se dispone de una base de datos de Datos de Consulta.

Escenario de éxito principal:

1. El Sistema almacena el listado de videos coincidentes en la base de datos de Datos de Consulta.

Extensiones: -

Postcondición: Se dispone de una base de datos de Datos de Consulta con una lista de videos que contienen coincidencias entre los términos de búsqueda y los contenidos de los subtítulos de cada video.

M4-Caso09: Guardar palabras vacías

Actor Principal: Módulo M4.

Activación: Módulo M4 busca coincidencias de lemas.

Precondición: Se dispone de un listado de frases.

Escenario de éxito principal:

1. Se procesa cada frase del listado término a término.
2. Se compara cada término con un listado de palabras vacías.
3. En caso de haber coincidencias, se elimina el término del listado de palabras.

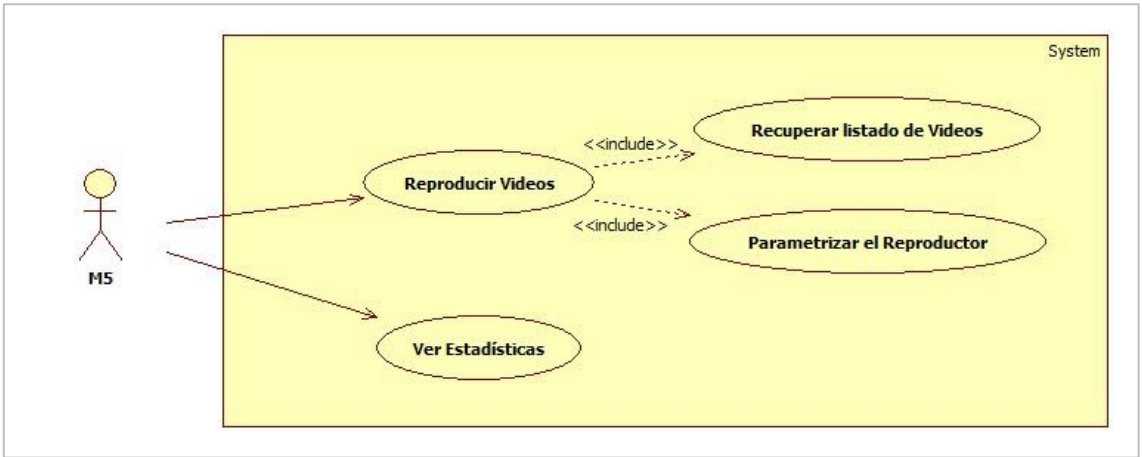
Extensiones: -

Postcondición: Se obtiene, por cada frase, un conjunto de términos sin palabras vacías.

5.2.6 Módulo M5

El módulo M5 se encarga de mostrar al usuario el resultado del proceso de búsqueda y análisis de datos.

5.2.6.1 Diagrama de casos de uso



.

Este resultado se muestra, por un lado, mediante un reproductor de los videos obtenidos, que se extrae de la base de datos de Datos de Consulta y que es el resultado de la ejecución de los módulos anteriores. Por otro lado, se muestran un conjunto de datos estadísticos resultantes del proceso de búsqueda, que pueden ayudar a entender mejor los datos obtenidos.

M5-Caso01: Reproducir Videos

Actor Principal: Módulo M5

Activación: Módulo M5 muestra el conjunto de videos resultante mediante un reproductor.

Precondición: Se ha obtenido, a través de la ejecución de los módulos anteriores, un listado de videos ubicado en la base de datos de Datos de Consulta con información relativa al video y a la marca de tiempo de cada uno de ellos.

Escenario de éxito principal:

1. El Sistema recupera el listado de videos resultante.
2. El Sistema parametriza el reproductor.
3. El Sistema reproduce los fragmentos en el reproductor de videos.

Extensiones:
1a: El listado de videos está vacío.
1. Se muestra un mensaje al usuario informando del suceso.

Postcondición: Se obtiene una reproducción de los fragmentos de video resultantes del proceso de búsqueda y análisis.

5.2.6.2 Descripción textual de Casos de Uso

M5-Caso02: Recuperar listado de Videos

Actor Principal: Módulo M5

Activación: Módulo M5 muestra el conjunto de videos resultante mediante un reproductor.

Precondición: Se ha obtenido, a través de la ejecución de los módulos anteriores, un listado de videos ubicado en la base de datos Datos de Consulta con información relativa al video y a la marca de tiempo de cada uno de ellos.

Escenario de éxito principal:

1. El Sistema realiza una consulta de la base de datos Datos de Consulta para recuperar el listado de videos.
2. El Sistema genera un fichero de reproducción, a partir de los resultados obtenidos, que cumple el formato requerido por el reproductor.
3. El Sistema almacena este fichero en una ruta local.

Extensiones:
1ª: El listado de videos está vacío.

1. El Sistema genera el fichero de reproducción sin ningún elemento.

Postcondición: Se obtiene un archivo de reproducción compatible con el reproductor del listado de videos compatible con la búsqueda.

M5-Caso03: Parametrizar el Reproductor

Actor Principal: Módulo M5

Activación: Módulo M5 muestra el conjunto de videos resultante mediante un reproductor.

Precondición: Se dispone de un fichero de reproducción de los videos resultantes.

Escenario de éxito principal:

1. El Sistema configura el Reproductor estableciendo el fichero de lista de reproducción de videos obtenido.

Extensiones: -

Postcondición: Se obtiene una reproducción de los fragmentos de video resultantes del proceso de búsqueda y análisis.

M5-Caso04: Ver Estadísticas

Actor Principal: Módulo M5

Activación: Módulo M5 muestra estadísticas del proceso de búsqueda.

Precondición: Se dispone de una base de datos de Datos de Consulta con información estadística del proceso de búsqueda y análisis.

Escenario de éxito principal:

1. El Sistema realiza una consulta de los datos estadísticos del proceso de búsqueda y análisis en la base de datos de Datos de Consulta.
2. El sistema muestra los datos obtenidos a través de la interfaz de la aplicación.
3. El sistema muestra también un listado del conjunto de videos extraídos, junto con un enlace a la fuente original del video del que se ha extraído cada fragmento.

Extensiones: -

Postcondición: Se muestran al usuario un conjunto de datos estadísticos del proceso de búsqueda y análisis.

5.2.7 Análisis dinámico

A continuación se exponen tres diagramas de secuencia correspondientes a los casos de uso de los módulos M3 y M4 que reflejan algunas de las acciones principales a ejecutar por la aplicación VideoClipping. Estos tres diagramas se muestran con el objetivo de clarificar las acciones descritas en la descripción textual de los casos de uso correspondientes.

- **Buscar estructuras sintácticas:**

Este diagrama refleja la acción de la separación de estructuras sintácticas a través del análisis de los comentarios de cada vídeo. Para dicha acción se requiere de la presencia de un gestor de estructuras capaz de identificar las diferentes estructuras sintácticas, y de un gestor de bases de datos que ha de almacenar dichas estructuras en la base de datos de comentarios.

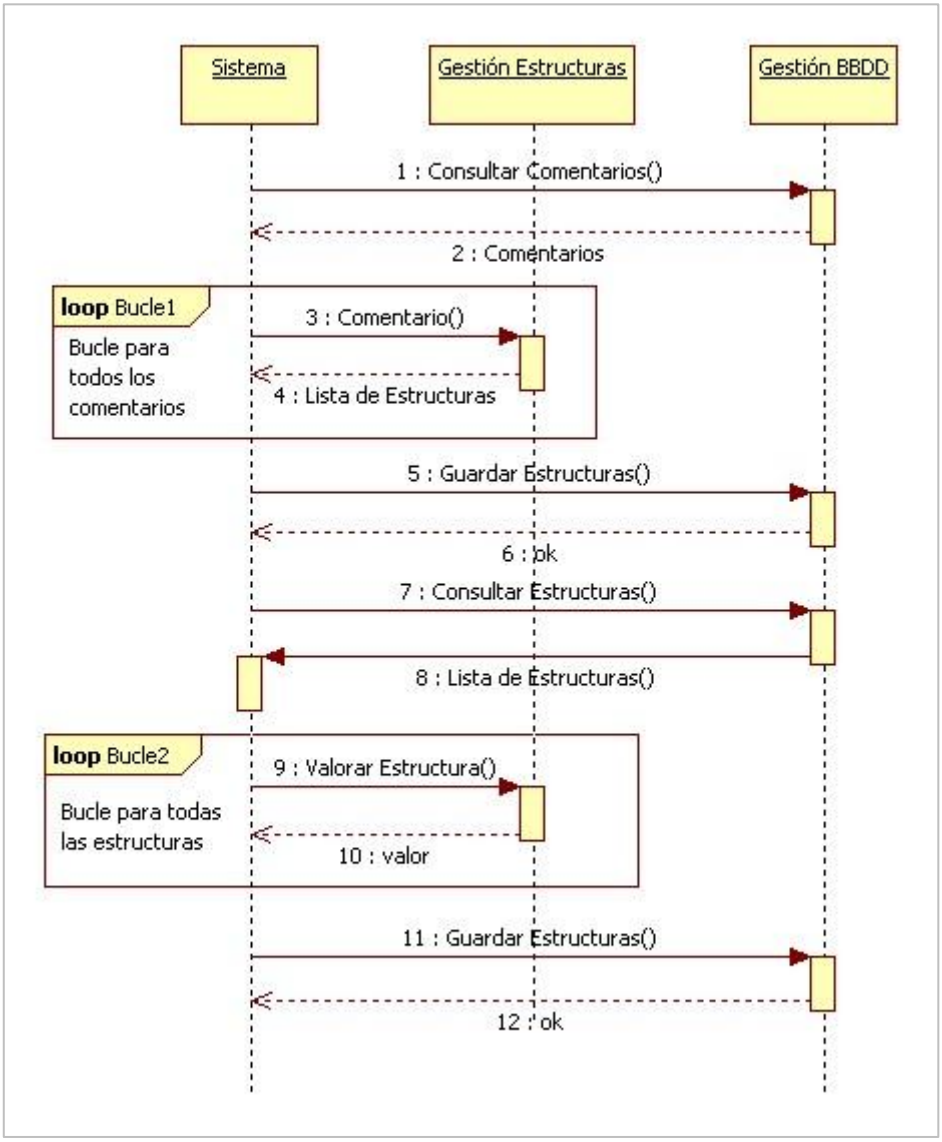


Ilustración 35: Diagrama de secuencia Buscar Estructuras Sintácticas.

- **Medir la relevancia:**

Mediante este diagrama se describe el proceso de cálculo de relevancia. Para ejecutar esta acción se requiere de un gestor de métricas que calcule, a partir de los datos estadísticos del vídeo, el valor de relevancia. Este valor es almacenado en la base de datos de vídeos para su posterior consulta.

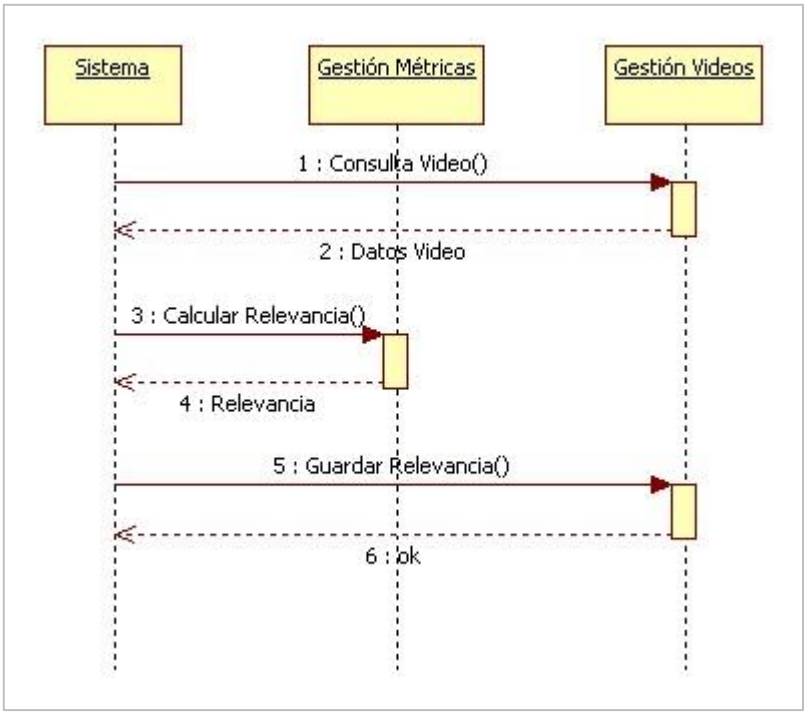


Ilustración 36: Diagrama de secuencia Medir la Relevancia.

- **Obtener sinónimos:**

La obtención de sinónimos, reflejada mediante el diagrama representado en la ilustración 37, muestra las acciones entre el sistema, el gestor de datos de la consulta, el gestor de ontologías y el lematizador.

En primer lugar el sistema obtiene los datos de la consulta mediante el gestor de consultas, que le retorna un listado de los términos que el usuario ha especificado en el filtro inicial. Posteriormente el sistema realiza una consulta al gestor de ontologías, enviándole el listado de términos introducidos por el usuario en el formulario de búsqueda. Este gestor analiza las palabras y retorna al sistema un listado de sinónimos de estos términos.

A continuación el sistema realiza una llamada al lematizador solicitando la obtención de los lemas del listado de términos, que contienen tanto los términos originales de la consulta como los sinónimos.

Finalmente, una vez recuperada la lista de términos lematizados, el sistema envía este listado al gestor de datos de la consulta, para almacenarlos en la base de datos correspondiente.

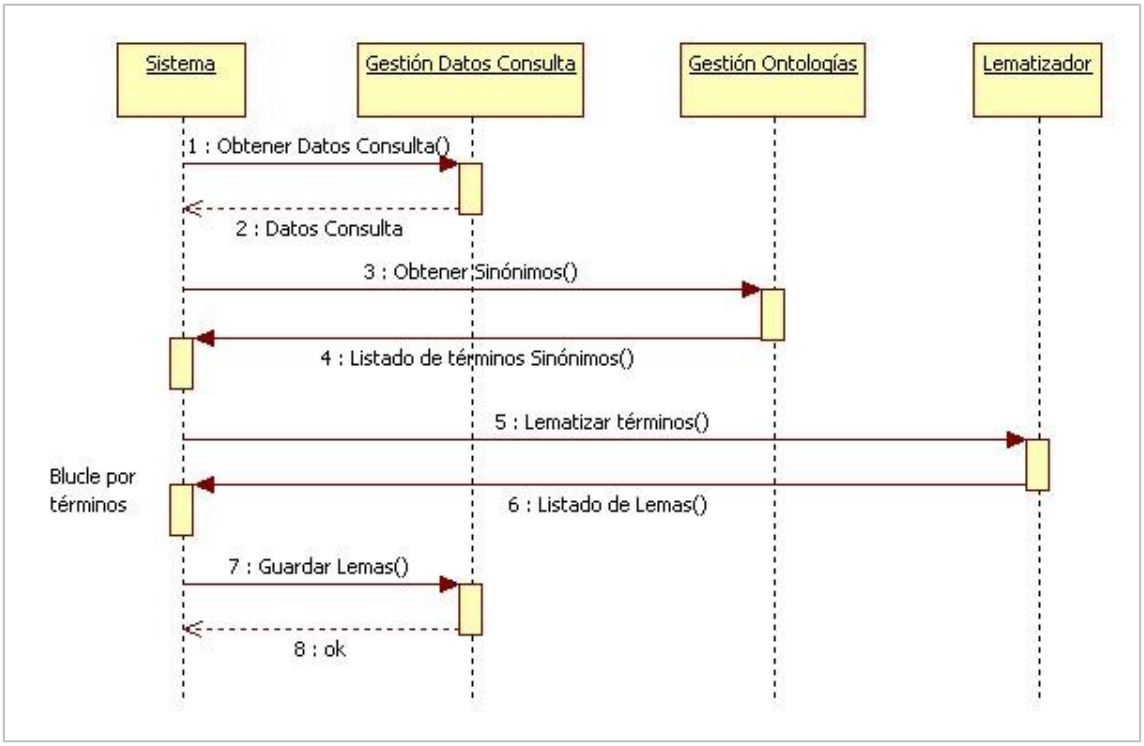


Ilustración 37: Diagrama de secuencia Obtener Sinónimos.

5.3 Arquitectura

A continuación se define la arquitectura necesaria para la aplicación VideoClipping teniendo en cuenta los requerimientos definidos en los apartados anteriores. Posteriormente se define el conjunto de tecnologías necesarias para implementación de la aplicación en un entorno real.

5.3.1 Diagrama de Despliegue

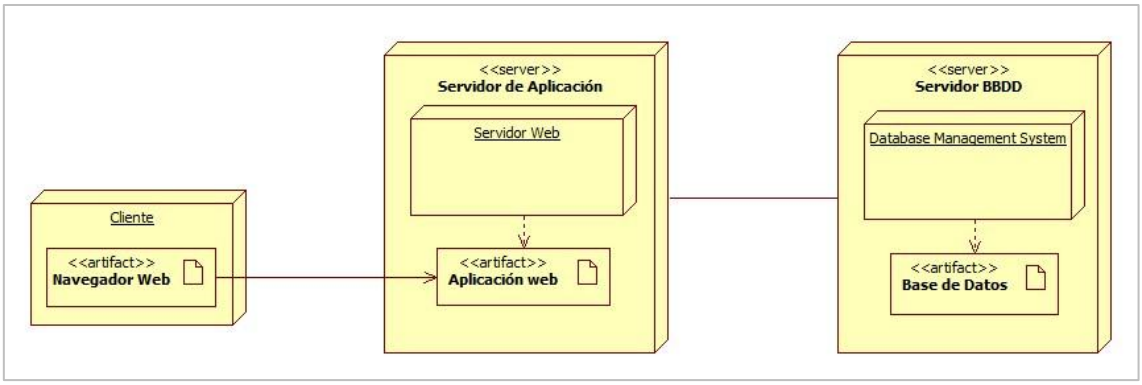


Ilustración 38: Diagrama de despliegue.

Como se puede observar, se ha organizado la arquitectura teniendo en cuenta las tres capas de presentación, lógica del negocio y lógica de datos. De este modo se dispone de una elevada escalabilidad, pudiendo modificar cada uno de los elementos de forma separada.

Además, esta distribución de los elementos permite poder tener servidores de prestaciones especialmente diseñadas para la tarea a realizar: las características necesarias del servidor de base de datos son diferentes a las del servidor de la aplicación.

5.3.2 Selección de las tecnologías

En cuanto a la elección de las tecnologías a utilizar es necesario tener en cuenta las diferentes funcionalidades anteriormente mencionadas, con el fin de seleccionar aquellas tecnologías que mejor se adapten a la solución propuesta.

Conviene tener en cuenta que la naturaleza de este proyecto, desarrollado en un marco de investigación, no establece limitaciones de mercado o requerimientos de cliente que acoten las posibilidades en este sentido. Por ello los factores determinantes y por ello motivos de elección tecnológica se basan siempre en dos principios:

- **Facilidad de implementación** de VideoClipping con respecto a otras alternativas existentes.

- **Optimización del rendimiento** de la aplicación en el entorno de producción teniendo en cuenta las funcionalidades requeridas y las prestaciones tecnológicas de la opción seleccionada.

- **La relación con el resto de elementos** tecnológicos seleccionados en el marco del proyecto.

Teniendo este aspecto en consideración, es necesario distinguir diferentes frentes a cubrir. A saber:

- **Tecnologías de Programación**: aquellas relativas a la elección del tipo de código de programación utilizado en la implementación del proyecto.

- **Tecnologías de Software**: engloban tanto las tecnologías de sistemas operativos, como la de aquellas aplicaciones necesarias para la ejecución del sistema VideoClipping.

- **Tecnologías de Bases de Datos**: relacionadas con los sistemas gestores de bases de datos que han de almacenar los datos a procesar o gestionar por parte de la aplicación.

- **Reproducción de Videos**: aquellos elementos necesarios, teniendo en cuenta los requerimientos de VideoClipping, para la reproducción del conjunto de videos resultante.

- **Otras tecnologías y algoritmos**: teniendo en cuenta el procesado de la información por parte de los módulos, aspectos teóricos a aplicar o algoritmos en función de las necesidades.

A continuación se describen las elecciones tecnológicas realizadas en cada caso:

Tecnologías de Programación:

Para poder escoger el tipo de tecnología de programación sobre el que desarrollar el proyecto es necesario en primer lugar entender que se pueden dividir las necesidades de programación en dos grandes apartados: por un lado aquellas relacionadas con el entorno en el que la aplicación será ejecutada, y por otro y más importante, aquellas que afectan a la ejecución de la aplicación en sí misma. Este último aspecto está condicionado por el tratamiento que hace VideoClipping de la información, al fundamentar una parte importante de su actividad en el tratamiento de expresiones sintácticas. Y es precisamente este punto uno de los elementos clave en la elección tecnológica a nivel de aplicación.

Analizando aquellos lenguajes de programación orientados al web, podemos destacar dos grandes actores: .NET y Java.

.NET ofrece, entre otras características, una administración del código para la ejecución en un entorno más seguro, o desarrollar en diferentes lenguajes (siempre compatibles con su Framework) al realizar un compilado en código intermedio (MSIL[18]) que luego genera código específico para la máquina del entorno de desarrollo. De este modo, mejora también el rendimiento de la aplicación al ser adaptado a cada plataforma.

Sin embargo, aspectos como el sistema automático de gestión de la memoria (denominado *Garbage Collector*), o la misma administración de la seguridad del código hacen que el consumo de los recursos del servidor sean más elevados en comparación con otros entornos.

Java, por su parte, ofrece una gran versatilidad en cuanto al tipo de plataforma sobre la que puede ser ejecutado, al proporcionar un entorno virtual sobre el que ejecutar las aplicaciones (Java Virtual Machine - *JVM*) adaptado a cada caso, además de simplificar el trabajo con metodologías orientadas a objetos y disponer de multitud de APIs que facilitan el proceso de desarrollo.

Si bien, su principal ventaja (el ser multiplataforma gracias a la *JVM*) es también una de sus grandes desventajas: al tener que interpretar los programas, la velocidad de ejecución no es la de un ejecutable convencional.

Por otro lado, otros lenguajes orientados a web como PHP no ofrecen unas características de rendimiento o de facilidad en cuanto a programación orientada a objetos comparables con los anteriores, por lo que quedan directamente descartados.

Sin embargo, entro todos los anteriores el aspecto diferenciador es las posibilidades de soporte al análisis sintáctico que estos lenguajes ofrecen, y en este sentido JavaCC es un elemento diferenciador.

JavaCC es una herramienta para la generación de analizadores sintácticos en código Java a partir de una gramática definida. Este elemento es de gran utilidad para VideoClipping, al poder asistir en el trabajo con estructuras sintácticas, ya en el reconocimiento de la relevancia de los comentarios a partir del análisis de su contenido o en la distinción de frases de los subtítulos.

Es por este motivo que la elección del tipo de lenguaje de programación se decanta hacia el uso de Java, tanto en lo que respecta a la programación de la

[18] Siglas de Microsoft Intermediate Language

aplicación de análisis como en su vertiente web, mediante el uso de páginas JSP y Servlets.

Tecnologías de Software:

En este punto, la elección del sistema operativo va relacionada con el tipo de lenguaje de programación seleccionado.

En los entornos de producción podemos distinguir dos Sistemas Operativos presentes en los servidores: Linux y Windows.

Se puede afirmar que Linux es más seguro que Windows, dado que los principales ataques a servidores se realizan sobre aquellos con el sistema operativo de Microsoft, en parte por la robustez y complejidad del sistema de seguridad que Linux ofrece.

Por otro lado, los servidores Linux son por norma general más rápidos gracias a la estabilidad de la plataforma, así como a la eficiencia del código fuente de este operativo, aspecto que se traduce en un incremento de la velocidad.

Además, el hecho de tener una mayor estabilidad y ser menos vulnerables a los ataques hace que los servidores basados en Linux requieran menos mantenimiento y sean, por lo tanto, los más económicos.

Sin embargo, el sistema operativo de Microsoft también tiene sus ventajas: es más sencillo desarrollar y aprender a gestionar las aplicaciones y los recursos en un servidor Windows, por lo que la cantidad de aplicaciones y recursos disponibles para este sistema es mayor.

Dada la independencia que brinda la programación en Java con respecto al sistema Operativo seleccionado, y teniendo en cuenta las ventajas de estabilidad que Linux ofrece, aspecto importante para la ejecución de VideoClipping, se ha decidido seleccionar éste sistema y descartar Windows.

Tecnologías de Bases de Datos:

Teniendo en cuenta las necesidades de la aplicación con respecto a la gestión de los datos, podemos destacar tres posibles sistemas gestores de Bases de Datos: Microsoft SQL Server, Oracle y MySQL.

SQL Server es un gestor orientado a la programación con tecnologías .NET que hace uso de un lenguaje de programación propio, llamado Transact SQL (T-SQL). Tiene soporte para transacciones, así como Stored Procedures (procedimientos ejecutables en el propio gestor), y dispone de un entorno gráfico de administración que hace que su gestión sea más sencilla que con otras alternativas.

Si bien es un sistema gestor muy potente, requiere también unas prestaciones de Servidor elevadas, y el hecho que use T-SQL hace más complejo en ocasiones el mantenimiento del sistema. Además, requiere de un sistema operativo Windows para su ejecución.

Oracle, por su parte, es un sistema gestor que también soporta transacciones, es estable y escalable, y al igual que ocurre con el lenguaje de programación de Sun Micrososystems (Java), este gestor es multiplataforma.

El principal inconveniente de Oracle es su precio, más elevado que el resto de alternativas, además de la necesidad de administración del gestor en función del Servidor en el que se aloja. Por otro lado, es complejo aprender su funcionamiento en profundidad y por tanto difícil poder extraer todas las prestaciones que ofrece.

Finalmente, MySQL es el sistema gestor de bases de datos más expandido en el entorno web. Se distribuye bajo licencia libre y destaca por su velocidad en la consulta de datos. Es sencillo instalarlo y configurarlo y es muy estable en cuanto a funcionamiento.

Si bien no es tan escalable como el resto de gestores, MySQL ofrece unas prestaciones que cubren las necesidades de VideoClipping y al su aprendizaje y administración es más sencillo que el resto de alternativas, por lo que se ha seleccionado como sistema gestor de Bases de Datos para a aplicación a desarrollar.

Reproducción de Videos:

Existen multitud de alternativas de componentes de reproducción de videos en el web: FLV player, OSM player, FlowPlayer, etc. Todos ellos tienen características similares, se integran en los sitios web como componentes mediante librerías asociadas y por lo general son de libre distribución.

Sin embargo, de entre ellos se puede destacar FlowPlayer, por las capacidades de configuración que permite, el gran número de formatos distintos de video capaz de reproducir, la descarga progresiva de los contenidos, la conectividad con YouTube, la posibilidad de control mediante JavaScript o el soporte de inserción de elementos como imágenes o publicidad incrustadas en el video.

Otras tecnologías y algoritmos

Finalmente, conviene destacar otras tecnologías que son necesarias para el desarrollo del proyecto, y que están relacionados con el tratamiento de la información que hace VideoClipping.

La API de Datos de YouTube y los mecanismos de comunicación con la misma, o la morfología de los elementos relacionados con los videos, son algunos ejemplos de componentes necesarios.

Por otro lados, las técnicas de minería de datos y de análisis de estructuras sintácticas como la Lematización (o Stemming) haciendo uso del algoritmo de Porter, o técnicas relativas a la evaluación de la aparición términos en documentos son técnicas importantes a aplicar durante el desarrollo de el proyecto.

De este modo, se pueden resumir las tecnologías seleccionadas mediante la siguiente tabla:

Tipo	Selección
Tecnologías de Programación	Lenguaje Java
Tecnologías de Software	Sistema Operativo Linux
Tecnologías de Bases de Datos	MySQL
Reproducción de Videos	FlowPlayer
Otras tecnologías y algoritmos	API YouTube, algoritmo de Porter, etc.

Tabla 29: Tipo de tecnologías.

6 Estudio económico

La valoración económica del presente trabajo se puede hacer teniendo en cuenta las áreas de trabajo siguientes:

- **Estudio:**

  El apartado de estudio comprende no sólo las técnicas de *Web Mining* y *Information Retrieval* aprendidas, sino la investigación acerca del funcionamiento de la API de datos de YouTube y la forma en que se pueden recuperar los elementos asociados a los videos, como los comentarios o los subtítulos

  Por otro lado, se incluye en este apartado el estudio de mercado de los diferentes repositorios de videos *online* y sus funcionalidades.

- **Análisis:**

  El apartado incluye las tareas relativas especificación de requerimientos, comprenden el análisis previo de las posibilidades de la aplicación en función de lo aprendido en el apartado de análisis.

  Por otro lado, incluye el estudio de las metodologías relativas a la documentación de los requerimientos, tanto en lo relativo a los apartados como en lo referente a aspectos a tener en cuenta en la calidad dels software.

  Finalmente, incluye la redacción del documento de especificación de requerimientos incluido en el apartado de Análisis y Diseño del prototipo VideoClipping.

- **Diseño:**

  El apartado de diseño engloba todas las tareas relativas a la construcción de los diagramas de casos de uso, así como a la descripción textual de los mismos, detallada en el Análisis de Requerimientos y diseño funcional. Por otro lado, quedan aquí incluido lo relativo al diseño de la arquitectura necesaria para el desarrollo de VideoClipping.

Teniendo lo anterior en cuenta, el tiempo dedicado a cada una de las partes ha sido el siguiente:

Área	Horas
Estudio	152 horas
Análisis	96 horas
Diseño	63 horas

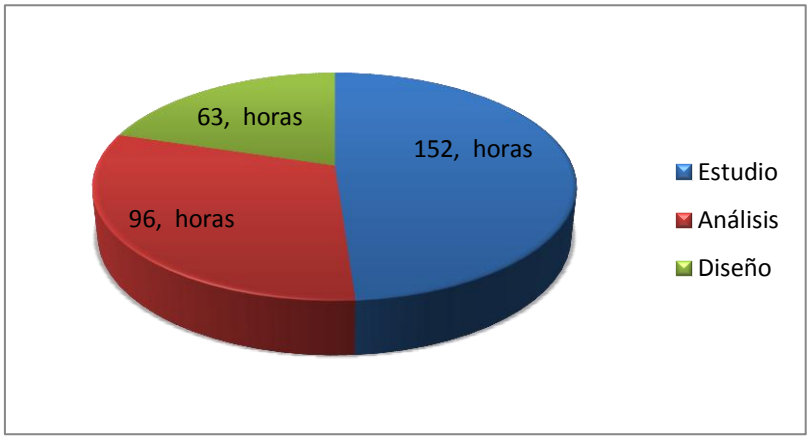


Ilustración 39: Distribución del tiempo.

7 Conclusiones y Líneas de Futuro

A continuación se detallan las conclusiones y las líneas de futuro extraídas a partir de la realización del proyecto.

7.1 Conclusiones

Las conclusiones extraídas engloban las diferentes áreas tratadas, por lo que se pueden dividir de la siguiente forma:

Web Mining y Information Retrieval:

Los campos de Web Mining y Information Retrieval engloban un conjunto de técnicas que tienen como objetivo la obtención de un conocimiento nuevo, desconocido hasta la fecha, a partir de un volumen de datos no estructurado.

Por lo que respecta al análisis de datos en el entorno web, el comportamiento del usuario es registrado en todo momento durante el proceso de navegación, y el análisis del conjunto de datos que deja tras de sí en un sitio puede ser analizado para mejorar la estructura del portal y la experiencia del cliente (o el usuario).

Desde otro punto de vista, se puede analizar el contenido de los diferentes elementos que forman un portal. Sin embargo, el análisis del contenido relativo a elementos multimedia como videos o audios está, a día de hoy, poco desarrollado.

En función del tipo de datos que se quiera analizar se ha de aplicar una técnica u otra, o la combinación de varias. El orden en que estas técnicas se aplican no es arbitrario, y siempre tiene que ver con el tipo de preprocesado y limpieza que hay que hacer de la información a analizar.

Por otro lado, existen varias técnicas de representación de la información que dependen también del objetivo final de la aplicación desarrollada. No hay una representación mejor que otra, todas tienen ventajas e inconvenientes en función del volumen de documentos a analizar, del tipo datos que estos documentos, o de cómo se organicen los datos dentro de los documentos.

Además, la representación escogida tiene que tener en cuenta el tipo de consulta que se haga posteriormente y las necesidades del usuario que la vaya a ejecutar. Por ejemplo, si se quiere obtener información acerca de la presencia de un término en un documento, o si además se quiere obtener un valor que indique la importancia relativa que tiene esa palabra en el mismo documento o con respecto a otros.

El proceso de Information Retrieval está compuesto por varias fases que dependen de forma directa al lenguaje con que estén escritas. Es decir, que las diferentes técnicas aplicadas en el proceso de IR se tienen que adaptar al idioma en que estén definidos los datos a analizar. Por lo tanto, hay dos maneras de lidiar con discrepancias en este sentido: cambiar los datos a analizar para hacer que coincidan con el tipo de lenguaje para el que están diseñados los algoritmos a aplicar, o bien definir diferentes algoritmos en función del idioma de los datos a procesar.

Repositorios de Videos online:

Los portales web de repositorios de video están de moda, y crecen tanto en número de usuarios como en contenidos que almacenan, ya sea videos, comentarios o otros elementos como subtítulos. Existen varios tipos de portales y la calidad de los mismos depende fundamentalmente de: el número de usuarios que los visitan y la autoría de los videos.

Los repositorios de video disponibles apenas ofrecen a día de hoy datos relativos a los subtítulos de los videos. Y si bien parece que esto tiene a cambiar, aún se contradice con el resultado de algunos estudios[19] que aseguran que un video recibe el 80% de visitas cuando contiene subtítulos que cuando no. YouTube ha añadido la recientemente posibilidad de subtitular, mediante transcripción automática, videos en inglés. Sin embargo esta funcionalidad no se aplica aún a la totalidad de videos del portal.

La API de datos de YouTube, se integra dentro del conjunto de librerías que Google pone a disposición de los usuarios para interactuar con sus datos. Para ello Google pone a disposición de los usuarios una documentación accesible a través del enlace de desarrolladores. Sin embargo, esta documentación no recoge todas las posibilidades de interacción con los elementos de youtube.

En el caso de los subtítulos, por ejemplo, la documentación informa de la posibilidad de recuperar los subtítulos de un video a través de la API, pero no indica la forma para hacerlo. Información recogida en algunos foros apunta a que, al ser (los subtítulos y la traducción automática) elementos relativamente nuevos, informarán de más métodos de consulta en nuevas versiones de la API.

Por otro lado, es muy difícil encontrar información relativa al proceso de búsqueda o tratamiento de la información que hace YouTube durante el proceso de búsqueda, para poderla integrar con el proceso de búsqueda de VideoClipping. En muchas ocasiones no se informa de todas las funcionalidades disponibles, ni del funcionamiento adecuado de todos los elementos. Un ejemplo de este hecho es la traducción automática de los subtítulos a través del parámetro *tlang* de la ruta de la consulta. No se informa de este atributo en la documentación, y por lo tanto es imposible conocer de qué modo poder modificar adecuadamente su valor (tipo exacto de valores válidos, etc.) para el desarrollo de VideoClipping.

Las consultas a la API de YouTube se pueden hacer tanto a través de código como a través de una parametrización de las rutas de consulta. El resultado obtenido es el mismo, y la única diferencia es el método de consulta. De entre las dos opciones, la consulta a través de la URL es mucho más cómoda, es más fácil cambiar valores y más sencillo poder ver los resultados en función de la consulta realizada.

Metodología de Análisis y Diseño:

El hecho de haber desarrollado el SRS ha posibilitado el estudio en profundidad de las diferentes funcionalidades del sistema mediante una metodología estandarizada. Este aspecto ha ayudado enormemente a la comprensión de la

[19] Estudio realizado por *eMarketer* en Marzo de 2009. (Fuente: http://www.emarketer.com/(X(1)S(33jocm55r4utiaqaee33hvy2))/Article.aspx?R=1007004&AspxAutoDetectCookieSupport=1 – Fecha de consulta: 23 de Mayo de 2011)

complejidad del sistema, al detallar entradas, salidas y funcionalidades de cada uno de los módulos que forman VideoClipping.

Con el análisis detallado de cada uno de los módulos, se ha podido abarcar toda la problemática relativa al flujo de datos de la consulta y del procesamiento de subtítulos y comentarios. Este análisis ha servido para poder tener una primera impresión del tipo de problemáticas con las que lidiar en el futuro con el diseño detallado e implementación de la solución diseñada de forma genérica en este proyecto.

Conclusiones acerca de VideoClipping:

Respecto a la aplicación en sí, gracias al desarrollo del documento de SRS, el análisis de requerimientos y el diseño funcional, se han podido obtener las conclusiones siguientes:

La calidad del contenido de los comentarios es un aspecto importante de cara a la valoración adecuada de los videos a través de Video Clipping. Sin embargo, el volumen de comentarios actualmente disponible en los videos de YouTube son, por norma general, de una calidad inferior a la esperada inicialmente, por lo que es necesario el estudio de algún tipo de elementos que puedan discriminar aquellos contenidos que no aporten valor al proceso de búsqueda y valoración.

La graduación de lo favorable o desfavorable que es un comentario con respecto al contenido del video es un aspecto importante a tener en cuenta, puesto que de ello depende la aparición del video analizado en la reproducción final. Por ello es importante centrar esfuerzos en la interpretación sintáctica del texto de los comentarios.

Del mismo modo, la elección de los atributos necesarios para el cálculo del factor de relevancia, así como los campos utilizados para calcularlo, son de vital importancia para el éxito del proyecto.

El uso de una metodología por módulos como la aplicada en este proyecto hace posible la ampliación del análisis de datos de videos a otros repositorios además de YouTube, por lo que se expanden las posibilidades de la solución propuesta.

7.2 Líneas de Futuro

En cuanto a las líneas de futuro, a continuación se expone un listado de diferentes alternativas de continuación del presente proyecto.

Tras el diseño general de la aplicación, la línea de futuro más clara es la realización de un diseño detallado y una implementación del sistema VideoClipping para una futura puesta en marcha de la solución en un entorno de producción.

La consulta mediante API de datos o mediante URL retorna el mismo tipo de datos. Sin embargo sería conveniente estudiar de forma empírica, mediante el desarrollo de un pequeño prototipo, el tiempo de respuesta de cada una de las consultas, con el fin de aplicar la más conveniente en la implementación final de VideoClippling.

El encontrar, a partir del análisis de requerimientos de la aplicación, que el análisis del contenido de los comentarios constituye un elemento de vital importancia para la aplicación propuesta, es importante analizar nuevas fórmulas de análisis que determinen el tipo de tratamiento a hacer del contenido de los mismos, así como el volumen de comentarios a recuperar para cada uno de los videos analizados.

El cálculo de la relevancia, del mismo modo importante para la solución final, podría ser revisado para incorporar datos relativos a otros videos relacionados con el video a tratar en cada momento. De este modo se ampliaría la búsqueda no sólo a los términos buscados o a sus sinónimos, sino a términos que si bien no pertenecen al conjunto anterior son también importantes en el resultado final y en el cálculo de un factor de relevancia del video.

Tal y como se apuntaba en las conclusiones, el hecho de haber realizado el diseño general de forma modular, permite la aplicación de partes de la solución propuesta a otros entornos. Estos nuevos entornos comprenden, no sólo otros repositorios de videos, sino otro tipo de aplicaciones como Twitter, donde el análisis de la información no tendría como resultado un conjunto de extractor de video, sino un conjunto de mensajes ordenados por relevancia, calculada en base a la valoración positiva o negativa de los mismos.

8 Bibliografía

Akerkar Rajendra y Lingras Pawan Building an Intelligent web [Libro]. - Sundbury, Massachussetts : Jones and Bartlett publishers, 2008.

Alag Stanam Collecting Intelligence in action [Libro]. - Greenwich : Manning Publications, 2009.

Arshad Salman Retrieve Title, Description and Thumbnail of a YouTube Video Using JavaScript and AJAX [En línea]. - 10 de Enero de 2010. - Marzo de 2011. - http://911-need-code-help.blogspot.com/2010/01/retrieve-youtube-video-title.html.

comScore, Inc. Más de 19 millones de internautas españoles vieron los 3.000 millones de videos online en Marzo 2011 [En línea] // YouTube vuelve a Liderar el Ranking con más de Tres horas por Individuo. - 27 de Mayo de 2011. - Mayo de 2011. - http://www.comscore.com/esl/Press_Events/Press_Releases/2011/5/Mas_de_19_millones_de_internautas_espanoles_vieron_los_3000_millones_de_videos_online_en_Marzo_2011.

Davenport Thomas H. y Prusak Laurence Working Knowledge: How organizations manage what they know [Libro]. - [s.l.] : Harvard Business Press, 2000.

Google Automatic Captions in YouTube Demo [En línea]. - 18 de Noviembre de 2009. - Marzo de 2011. - http://www.youtube.com/watch?v=kTvHIDKLFqc.

Google Code Developer's Guide (v2): Using REST [En línea]. - 2011. - Mayo de 2011. - http://code.google.com/intl/es-ES/apis/language/translate/v2/using_rest.html#query-params.

Google Code Developer's Guide: Data API Protocol – API Query Parameters [En línea]. - 2011. - Mayo de 2011. - http://code.google.com/intl/en-EN/apis/youtube/2.0/developers_guide_protocol_api_query_parameters.html#fmtsp.

Google Code Developers Guide: Data API Protocol – Captions [En línea]. - 2011. - Mayo de 2011. - http://code.google.com/intl/en-EN/apis/youtube/2.0/developers_guide_protocol_captions.html#Retrieve_Caption_Track_Translation.

Google Code Developer's Guide: PHP [En línea]. - 2011. - Mayo de 2011. - http://code.google.com/intl/es-ES/apis/youtube/1.0/developers_guide_php.html.

Google Code Guía de referencia: Protocolo de API de datos [En línea]. - 2010. - Abril de 2010. - http://code.google.com/intl/es-ES/apis/youtube/reference.html#Videos_feed.

Google Code Guía para desarrolladores: protocolo del API de datos – Parámetros de consulta del API [En línea]. - 2010. - Abril de 2011. - http://code.google.com/intl/es-ES/apis/youtube/2.0/developers_guide_protocol_api_query_parameters.html.

Google Code Guía para desarrolladores: protocolo del API de datos – Subtítulos [En línea]. - 2010. - Abril de 2011. - http://code.google.com/intl/es-ES/apis/youtube/2.0/developers_guide_protocol_captions.html#Retrieve_Caption_Track.

Google Code Reference Guide: Data API Protocol [En línea]. - 14 de Abril de 2011. - Mayo de 2011. - http://code.google.com/intl/en-EN/apis/youtube/2.0/reference.html.

Google Code YouTube Chromeless Player Example [En línea]. - 2011. - Abril de 2011. - http://code.google.com/intl/es-ES/apis/youtube/chromeless_example_1.html.

Harrenstien Ken Un año de subtítulos automáticos y más novedades [En línea]. - 26 de Noviembre de 2010. - Marzo de 2011. - http://youtube-espanol.blogspot.com/2010/11/un-ano-de-subtitulos-automaticos-y-mas.html.

Hartmann Jochen Getting Started with the Google Data PHP Client Library [En línea]. - Octubre de 2008. - Mayo de 2011. - http://code.google.com/intl/en-EN/apis/gdata/articles/php_client_lib.html.

Hear-it More and more hearing impaired people [En línea] // More than 700 million hearing-impaired people by 2015. - Febrero de 2011. - http://www.hear-it.org/page.dsp?area=134.

Hodgson Dom PHP Stop Word List [En línea]. - 2 de Mayo de 2010. - Mayo de 2011. - http://stackoverflow.com/questions/2752896/php-stop-word-list.

Javascript Porter Stemmer Online [En línea]. - 18 de Noviembre de 2010. - Abril de 2011. - http://qaa.ath.cx/porter_js_demo.html.

JSFiddle.net JSFiddle Alpha- Online web editor [En línea]. - 2010. - Abril de 2011. - http://jsfiddle.net/salman/bwzpD/12/.

LaVanguardia.com YouTube recibe cada minuto 48 horas de vídeo [En línea]. - 25 de Mayo de 2011. - Mayo de 2011. - http://www.lavanguardia.com/internet/20110525/54160394044/youtube-recibe-cada-minuto-48-horas-de-video.html.

Miller Michael YouTube for business [Libro]. - Indiana : Que Publishing, 2010.

Nadeau David R. PHP tip: How to extract keywords from a web page [En línea]. - 13 de Abril de 2008. - Abril de 2011. - http://nadeausoftware.com/articles/2008/04/php_tip_how_extract_keywords_web_page#Stemthewords.

php.net SimpleXML [En línea]. - 27 de Mayo de 2011. - Mayo de 2011. - http://php.net/manual/es/book.simplexml.php.

php.net stristr [En línea] // (PHP 4, PHP 5). - Mayo de 2011. - Mayo de 2011. - http://es2.php.net/stristr.

Porter Martin Spanish stemming algorithm [En línea]. - 15 de Junio de 2005. - Abril de 2011. - http://snowball.tartarus.org/algorithms/spanish/stemmer.html.

Posnick Jeffrey YouTube Captions Uploader Web App [En línea]. - 27 de Enero de 2011. - Abril de 2011. - http://apiblog.youtube.com/2011/01/youtube-captions-uploader-web-app.html.

SiteFuse MrPhear - PHP stop words function [En línea]. - Agosto de 2010. - Mayo de 2011. - http://www.sitefuse.com/forums/thread/1-php-stop-words-function.

Sourceforge.net ¿Qué es Google2SRT? [En línea]. - Marzo de 2011. - http://google2srt.sourceforge.net/es/index.html.

Sourgeforce.net Stemmer-es [En línea] // A spanish stemmer / Un lematizador de español . - Abril de 2011. - http://stemmer-es.sourceforge.net/.

Surinder's blog Use YouTubes RSS Feed To Output A List of Videos [En línea]. - 18 de Junio de 2009. - Mayo de 2011. - http://surinder.computing-studio.com/post/2009/06/18/Use-YouTubes-RSS-Feed-To-Output-A-List-of-Videos.aspx.

Tokusei Hiroto The Future Will Be Captioned: Improving Accessibility on YouTube [En línea]. - 4 de Marzo de 2010. - Febrero de 2011. - http://youtube-global.blogspot.com/2010/03/future-will-be-captioned-improving.html.

TopECB InfoKeko - usuario Ejemplo de uso de API de datos (GDATA) de YouTube [En línea]. - 11 de Octubre de 2009. - Mayo de 2011. - http://www.topecb.es/source_youtube.php.

Vaswani Vikram Use the YouTube API with PHP [En línea] // Process and integrate data from YouTube into your PHP application with PHP's SimpleXML extension. - 15 de Abril de 2008. - Marzo de 2011. - http://www.ibm.com/developerworks/xml/library/x-youtubeapi/.

Velásquez Juan D. y C.Jain Lakhmi Advanced Techniques in Web Intelligence [Libro]. - Santiago de Chile : Springer-Verlag Berlin Heidelberg, 2010.

West Ron YouTube API: Dive In [En línea]. - 2009. - Abril de 2011. - http://www.slideshare.net/notronwest/youtube-api-dive-in.

YouTube Help Adding and Editing captions / subtitles [En línea]. - 28 de Enero de 2011. - Marzo de 2011. - http://www.google.com/support/youtube/bin/answer.py?hl=en&answer=100077.

9 Anexo: Tabla de ilustraciones