UNIVERSITAT POLITÈCNICA DE VALÈNCIA
ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA
DEPARTAMENTO DE SISTEMAS INFORMÁTICOS Y COMPUTACIÓN

UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

Integración en Matterhorn de técnicas de asistencia a la transcripción de audio.

Proyecto final de carrera - Ingeniería Informática

Alejandro Pérez González de Martos

Supervisado por:
Dr. Alfons Juan-Císcar
Dr. Jorge Civera Saiz

19 de julio de 2012

Índice general

Capítulo 1

Introducción

Los repositorios de contenido multimedia de ámbito académico están creciendo rápidamente debido al progresivo avance de las nuevas tecnologías en nuestra sociedad. Charlas, conferencias o clases son grabadas desde cualquier parte del mundo y publicadas para ofrecer a aquellos que lo deseen la posibilidad de visualizar su contenido. En la era de la información, las universidades se adaptan al hoy en día medio de comunicación por excelencia, Internet. Universidades a distancia, como la UNED[1], clases no presenciales, pizarras digitales o plataformas de teleformación son sin duda un buen ejemplo de ello. Es fundamentalmente por este motivo por el que los repositorios online comienzan a albergar grandes cantidades de contenidos educativos, y por el que la cantidad de información disponible crece de manera exponencial a medida que las principales entidades emisoras, en este caso las universidades, adoptan políticas de libre distribución de contenidos.

El principal problema con el que nos encontramos a la hora de facilitar el visionado de estos contenidos al mayor número de personas posible es la variedad lingüística existente. Aunque el inglés se consolida cada vez más como el idioma oficial en el mundo globalizado en el que vivimos, los contenidos educativos de los distintos repositorios existentes suponen una gran variedad de lenguas. Sin embargo, la mayoría de la información disponible en estos repositorios no está ni transcrita ni traducida a distintos idiomas, lo que además de solventar las diferencias lingüísticas facilitaría el visionado de los contenidos a personas con discapacidad. Debido a la gran cantidad de contenidos disponible y a su rápido crecimiento, la transcripción y traducción manuales suponen un tremendo coste y esfuerzo, lo que las hace prácticamente inviables.

Es por ello que la transcripción automática o reconocimiento automático del habla ha adquirido vital importancia, puesto que supone el paso previo a la traducción automática. El reconocimiento de formas es la disciplina dentro del área de la Inteligencia Artificial donde se engloba el reconocimiento automático del habla. Este último sigue siendo hoy en día uno de los principales focos de investigación a nivel mundial debido a la gran complejidad que supone. Actualmente, los sistemas de reconocimiento automático del habla con mayor éxito se basan en redes de estados finitos estocásticas.

[1] UNED, *Universidad Nacional de Educación a Distancia.*

Sin embargo, los resultados obtenidos para vocabularios amplios no son lo suficientemente precisos como para que resulten útiles por si mismos. Hablando en términos generales, de un buen sistema de reconocimiento automático del habla para este ámbito podemos esperar de un 20 a un 40 por ciento de palabras mal reconocidas. Es necesaria por tanto la intervención humana para mejorar la calidad de las transcripciones ofrecidas por el sistema de transcripción automática y obtener transcripciones suficientemente precisas.

Aun así, y teniendo en cuenta la ingente cantidad de información existente y generada día a día que necesita de ser transcrita y traducida, la revisión y corrección puramente manual de transcripciones generadas automáticamente supone una tarea cuanto menos tediosa. Con el fin de proporcionar una solución efectiva y práctica al problema de la transcripción semi-automática, se estudiará la creación de un sistema inteligente de edición interactiva de transcripciones.

1.1. Reconocimiento automático del habla

1.1.1. Historia y evolución

Durante las últimas décadas, la investigación en el campo del reconocimiento automático del habla se ha venido desarrollando de una forma intensa, empujada por los avances en procesamiento de señal, algoritmos, arquitecturas y plataformas de cómputo. Durante este periodo se han construido sistemas para una amplia gama de aplicaciones, que abarcan desde tareas de reconocimiento de pequeños conjuntos de palabras sobre líneas telefónicas, hasta máquinas de dictado para grandes vocabularios con capacidad para asimilar cualquier tipo de habla [LRRL96]. La historia del campo de investigación del reconocimiento automático del habla se ha venido llevando a cabo desde la segunda mitad del siglo XX.

Los primeros intentos por construir máquinas que realizaran tareas de reconocimiento se remontan a la década de los 50, cuando diversos investigadores trataban de explotar los principios fundamentales de la fonética acústica. En 1952, en los laboratorios Bell, K. Davis, R. Biddulph y S. Balashek crearon un sistema electrónico que permitía identificar para un solo hablante, pronunciaciones de los 10 dígitos realizadas de forma aislada [KHDB52]. En 1959, en la University College de Londres, P. Denes trataba de desarrollar un sistema para reconocer 4 vocales y 9 consonantes [RJ93]. El aspecto más novedoso de su trabajo era el uso de información estadística, acerca de las secuencias válidas de fonemas en inglés. Sin embargo, todos estos experimentos corresponden a dispositivos electrónicos.

Los primeros experimentos de reconocimiento desarrollados en ordenadores tienen lugar al final de los años 50 y comienzo de los 60, principalmente en el Lincoln Laboratory a cargo de J. Forgie y C. Forgie [Jua98]. Es en la década de los 60 cuando se generaliza el uso de ordenadores en el campo del reconocimiento del habla. Durante estos años se inician varios proyectos que encarrilan la investigación en esta área. Los años 70 representan un periodo muy activo para esta disciplina, distinguiéndose dos actividades principales:

- Reconocimiento de palabras aisladas.
- Reconocimiento del habla continua.

Se comienzan a obtener buenos resultados en el reconocimiento de palabras aisladas, y su uso comienza a ser viable en la práctica. En cuanto al habla continua, los primeros trabajos cubrían el reconocimiento de oraciones de un mismo locutor con un vocabulario aproximado de 1.000 palabras. Es a raíz de ésto que se advierte que el conocimiento sintáctico, semántico y contextual son fuentes de información. El sistema Hearsay I, construido por la CMU[2] en 1973 era capaz de emplear información de tipo semántico para reducir el número de posibles alternativas que el reconocedor debía evaluar [Jua98]. En los AT&T Bell Labs los investigadores comenzaron una serie de experimentos orientados a conseguir reconocedores realmente independientes del locutor para su uso en aplicaciones telefónicas [Jua98].

A finales de los 70 y en los años 80 se progresa notablemente en la generalización en la construcción de sistemas de reconocimiento. Es aquí donde se produce un giro metodológico al pasar de métodos basados en comparación de plantillas a métodos basados en el modelado estadístico debido a la extensión en el uso de los Modelos Ocultos de Markov[3] [RJ86] [Rab89] que es el modelo usado en la actualidad para capturar y modelar la variabilidad existente en el habla. Durante este mismo periodo, el programa DARPA[4], impulsó en Estados Unidos el desarrollo de mejores sistemas de reconocimiento para habla continua y vocabularios de tamaño medio y grande con independencia del locutor.

Muchas de las contribuciones durante este periodo y el principio de los años 90, provienen de los esfuerzos de la CMU a través de su sistema SPHINX[Lee89]. La década de los 90 supone en cierta manera la continuidad en los objetivos ya propuestos, ampliando eso sí, el tamaño de los vocabularios a la vez que se diversifican los campos de aplicación.

1.1.2. Actualidad

En estos últimos años ha crecido el interés por el estudio de los procesos de reconocimiento en condiciones de ruido y adversas en general. Se intentan implementar nuevas técnicas que mejoren los resultados del reconocimiento, pues las soluciones existentes en la actualidad todavía no alcanzan un grado de precisión elevado para vocabularios con gran número de palabras. Parece que la evolución en el ámbito del reconocimiento del habla durante los últimos años ha sido mediante la acumulación de pequeñas mejoras que suponen tareas mucho más complejas, debido principalmente al aumento notable en la velocidad de cálculo y a ordenadores más potentes en general. La investigación y desarrollo actual se centran principalmente en estos tres puntos:

- Mejorar la robustez de los sistemas de reconocimiento del habla, no solamente ante el ruido sino ante cualquier condición que suponga la degradación del rendimiento del sistema.

[2] CMU, *Carnegie Mellon University.*
[3] En inglés *Hidden Markov Models (HMM).*
[4] DARPA, *Defence Advance Research Agency.*

- Debido a que existe gran cantidad de datos de habla humana, y a que sería demasiado costoso transcribir manualmente tanta cantidad de información, la investigación se centra en desarrollar nuevos métodos de aprendizaje automático que puedan emplear con cierta efectividad grandes cantidades de datos sin etiquetar mediante aprendizaje no supervisado para la mejora del sistema.
- Por último, mejorar el entendimiento de la capacidad humana para comprender el habla, y emplear esta información para mejorar el rendimiento de los sistemas de reconocimiento automático del habla.

1.2. Visión general del proyecto

Este proyecto está orientado a la integración en Opencast Matterhorn de técnicas interactivas para la transcripción asistida de audio. El principal objetivo del proyecto es llevar a cabo esta integración de acuerdo con los principios que guían el diseño de la plataforma Matterhorn. Matterhorn es una plataforma open-source para la gestión y administración de contenidos multimedia de ámbito académico. Las instituciones emplean Matterhorn para la grabación de lecciones, gestionar los vídeos existentes, publicar los mismos en distintos medios de distribución y proporcionar una interfaz de usuario para facilitar su visionado. La intención será pues integrar un módulo de reconocimiento del habla que permita obtener de manera semi-automática transcripciones de los vídeos añadidos al sistema.

Para llevar a cabo la construcción de un sistema de reconocimiento automático del habla es necesario disponer de un numeroso conjunto de datos, en nuestro caso vídeos educativos y sus transcripciones. Trabajaremos con el repositorio poliMedia [dV12]. PoliMedia es un servicio reciente pensado para la creación y distribución de material educativo en la Universidad Politécnica de Valencia. Actualmente contiene alrededor de 6.000 vídeos correspondientes a más de 1.000 horas de grabación. Emplearemos este corpus para la implementación de un software de reconocimiento que permita obtener transcripciones automáticas para los vídeos del mismo. Sin embargo, no entraremos en mucho detalle acerca del diseño de este sistema puesto que no es objeto de este proyecto.

Una vez seamos capaces de obtener transcripciones automáticas de cierta precisión para los vídeos del repositorio, estudiaremos la manera de editarlas interactivamente. Construiremos un reproductor prototipo basado en HTML5 sobre el que se irán implementando nuevas funcionalidades propias de la edición y corrección interactiva de transcripciones. Discutiremos el formato que deberán seguir las transcripciones en el sistema. En la actualidad el formato más extendido sea posiblemente SubRip[5], soportado por la mayoría de reproductores multimedia que permiten la visualización de subtítulos. Las transcripciones manuales de las que disponemos para el repositorio poliMedia siguen el formato empleado por Transcriber[6], que se asemeja al de un documento XML pero no permite la ampliación del mismo con nuevas etiquetas. Puesto que ni SubRip ni Transcriber permiten ser ampliados sin comprometer la

[5] SubRip, extensión .srt.
[6] Transcriber, extensión .trs. *http://trans.sourceforge.net.*

compatibilidad de los mismos, necesitaremos un formato más flexible que permita la representación de cierta información relevante como por ejemplo medidas de confianza[7] o indicar que cierto tramo de la transcripción ha sido modificado manualmente. También hablaremos del rol que los distintos usuarios tendrán en el sistema. Debemos tener en cuenta que el usuario común está interesado en el visionado del contenido, por lo que supondremos que no dedicará mucho tiempo a la corrección de transcripciones defectuosas. Se implementará un servicio web encargado de gestionar la comunicación entre el editor y el sistema de transcripción automática. El reproductor hará uso de dicho servicio web tanto para obtener las transcripciones de un vídeo determinado como para gestionar las modificaciones por parte del usuario en las transcripciones.

Por último veremos una propuesta de integración de este sistema de transcripción semi-automática en la plataforma Matterhorn con el fin de obtener un sistema evaluable en un entorno real y posibilitar su implantación en otros repositorios relacionados con Matterhorn.

[7] Medidas de confianza, en inglés: confidence measures.

Capítulo 2

Opencast Matterhorn

2.1. Introducción y características

Aunque se ha hecho una breve presentación en la introducción, vamos a conocer con detalle la plataforma Matterhorn, base de este proyecto. Matterhorn podría resumirse en pocas palabras como un "software para la grabación y gestión de contenidos audiovisuales en red para instituciones educativas".

Matterhorn es una plataforma gratuita, de código abierto, que da soporte a la gestión de contenido audiovisual en el ámbito académico. Las instituciones usarán Matterhorn para la grabación automatizada de sus clases o conferencias, gestionar las grabaciones existentes, servir dichas grabaciones a distintos canales de distribución y proporcionar una interfaz de usuario que posibilite el visionado de los contenidos a los estudiantes.

El proyecto Opencast Matterhorn pretende desarrollar un sistema LCDS (Lecture Capture and Distribution System) de segunda generación. Un sistema diseñado por universidades que ya han desarrollado y tienen en producción su propia plataforma LCDS y que, con la experiencia ganada, quieren crear entre todas una nueva plataforma con unas capacidades que ninguna sería capaz de alcanzar por separado y que tampoco se encuentran en los sistemas comerciales. Las principales características proporcionadas por la plataforma Matterhorn son las siguientes:

- Herramientas administrativas para grabaciones automatizadas, subida manual de archivos y gestión de metadatos y funciones de captura y procesamiento.
- Especificaciones de agentes de captura recomendados.
- Integración con los dispositivos de grabación existentes en las aulas para la gestión de capturas automatizadas.
- Servicios de procesamiento y codificación que preparan los archivos multimedia según especificaciones configurables.

- Distribución a difusores locales y servidores de descarga y capacidad de configuración para la distribución a distintos canales como YouTube, iTunes o a otros sistemas de gestión de contenidos.
- Una interfaz de usuario trabajada para sumergir a los estudiantes en el contenido, incluyendo visualización de diapositivas, búsqueda basada en el contenido y subtitulado.

2.2. Descripción del sistema

Matterhorn proporciona un framework de servicios para la gestión de vídeos académicos, que las instituciones pueden personalizar para alcanzar sus necesidades individuales. La arquitectura de Matterhorn consta de cuatro grandes bloques: captura y administración, ingesta y procesado, distribución y herramientas de acceso y participación. En la figura 2.1 podemos ver de manera resumida qué tareas se engloban en cada uno de estos bloques y cómo están relacionados entre sí.

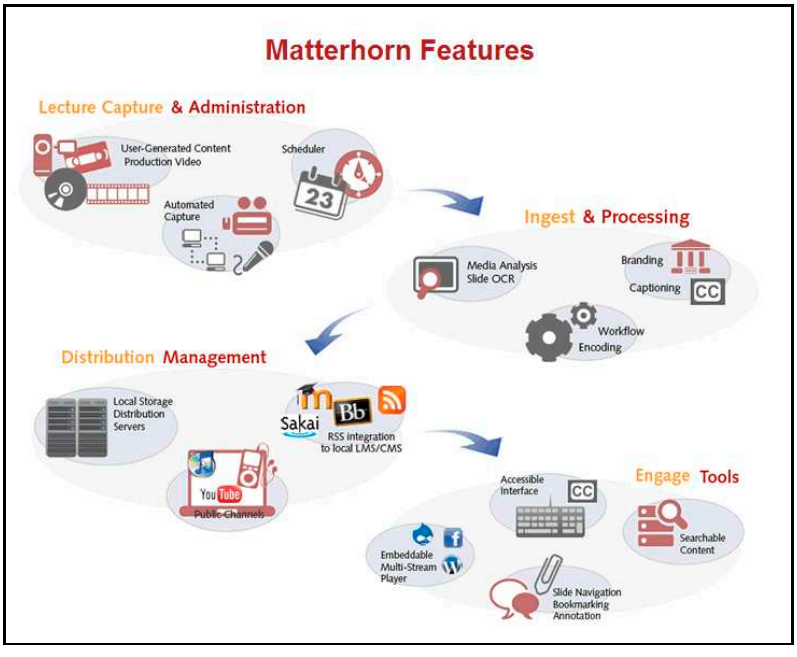

Figura 2.1: Ilustración de las características de Matterhorn.

La figura anterior muestra también el flujo de trabajo de Matterhorn, desde la captura de vídeo hasta la difusión a distintos canales de distribución. Describiremos brevemente las funcionalidades que aporta cada bloque del sistema:

1. **Captura y administración**: El proceso de grabación empieza por determinar qué es lo que debe ser grabado, cómo y de qué forma. Matterhorn ofrece la

posibilidad de automatizar las grabaciones. Se requerirá determinar información relativa a la grabación como fechas, hora, lugar o profesor. Por ejemplo, es posible programar una captura periódica para "Aula 1.E, todos los Martes de 10:00 a 12:00, Derecho Mercantil, Profesor Juan Carlos Gómez".

2. **Ingesta y procesado**: Al finalizar las grabaciones éstas son enviadas a una bandeja de entrada para ser procesadas. Las diferentes pistas de la grabación (audio, diapositivas, vídeo) son procesadas siguiendo un determinado *workflow* y posteriormente "empaquetadas" en lo que se conoce como *mediaPackage*. Un mediaPackage es un documento XML que incluye metadatos simples (como por ejemplo un identificador, un título, etc.) y una lista de los elementos "empaquetados": principalmente las pistas de audio y vídeo, ficheros de metadatos y otros archivos adjuntos.

3. **Distribución**: La distribución de los contenidos varía enormemente entre universidades: van desde la simple integración de los vídeos en su WCMS[1] local hasta distribuir vía iTunes o YouTube. El módulo de distribución gestiona todas estas necesidades de un modo configurable.

4. **Herramientas de acceso y participación**: Engloba distintas aplicaciones como el reproductor de contenidos multimedia, la búsqueda basada en contenidos, una interfaz de accesibilidad, etc.

Matterhorn está basado en Java como lenguaje de programación para crear las aplicaciones necesarias y una infraestructura SOA[2]. El diseño fundamental del programa radica en la modularización de sus componentes y se basa en la tecnología OSGi[3].

La plataforma de servicios OSGi proporciona un estándar para los entornos orientados a componentes para la cooperación de servicios de red [SH07]. Su objetivo es definir las especificaciones de software que permitan diseñar plataformas compatibles que puedan proporcionar múltiples servicios. Las aplicaciones o componentes en una plataforma OSGi pueden ser instalados, iniciados, detenidos, actualizados y desinstalados de manera remota y sin necesidad de reiniciar la aplicación. Matterhorn usa la implementación de código abierto de OSGi R4 Apache Felix [Apa].

Nos encontramos por lo tanto ante un sistema completamente modularizado, orientado a componentes y servicios. Matterhorn, en su versión 1.3, está compuesto por más de 60 módulos que interactúan a través de más de 20 servicios. En el marco de la plataforma OSGi, los módulos son comunmente llamados *bundles*. Cada módulo o bundle tiene una función específica dentro del sistema. Por citar algunos ejemplos, tenemos el módulo de captura "matterhorn-caption", el módulo de ingesta de vídeos "matterhorn-ingest", el módulo encargado de la segmentación de vídeo "matterhorn-videosegmenter", el módulo de reconocimiento OCR[4] "matterhorn-textanalyzer" y un largo etcétera.

[1] Web Content Management System.
[2] Service Oriented Architecture, en español: Arquitectura Orientada a Servicios.
[3] Open Services Gateway initiative.
[4] Optical Character Recognition, en español: Reconocimiento Óptico de Caracteres (ROC).

2.2.1. Matterhorn mediaPackage

El elemento básico de información dentro de la plataforma Matterhorn se conoce como *mediaPackage*. Un mediaPackage consiste en un documento XML que incluye metadatos básicos (un identificador, título, etc.) y una lista de los elementos que contiene, que se dividen tres tipos:

- Media tracks (pistas de audio/vídeo)
- Metadata catalogs (catálogos de metadatos)
- Attachments (archivos adjuntos/otros)

Normalmente en el ámbito académico un mediaPackage estará formado por una o dos pistas acompañadas de catálogos de metadatos, y en algunos casos de las diapositivas correspondientes. El proceso de añadir las grabaciones realizadas al sistema termina por tanto, entre otras cosas, en la creación de un mediaPackage que contenga los elementos necesarios para su almacenamiento y archivación.

Este proceso se realiza automáticamente al añadir nuevos elementos al sistema (ver figura 2.2). Primero, a través del IngestService se creará un mediaPackage vacío mediante la llamada *createMediaPackage*, que generará también su identificador. Además se creará un directorio en el sistema para alojar los elementos que serán añadidos en las siguientes llamadas al IngestService. Se crearán las "etiquetas" de los elementos que contendrá el mediaPackage mediante las llamadas *addMediaPackageTrack*, *addMediaPackageCatalog* y *addMediaPackageAttachment*. A continuación se descargarán los archivos desde la bandeja de entrada y se añadirán al *WorkingFileRepository* para hacerlos accesibles localmente. Una vez descargados los archivos de medios, estos son inspeccionados por el MediaInspectionService que extraerá información técnica de los mismos (duración, códecs, etc.) para añadirla al mediaPackage. Llegados a este punto, el mediaPackage está totalmente compilado. En este momento será recogido por el ConductorService, que será el encargado de decidir qué hacer a continuación con este mediaPackage. El ConductorService se encargará fundamentalmente de procesar el mediaPackage a través de una serie de operaciones (principalmente llamadas a otros servicios) que incluyen desde la aplicación de distintos códecs a los medios, a la archivación y publicación de los mismos en distintos canales de distribución. Esta serie de operaciones es también conocida como *workflow*.

2.2.2. Matterhorn workflow

Un workflow en Matterhorn consiste en un documento XML que contiene una lista ordenada de operaciones. No hay límite en el número de operaciones ni en el número de repeticiones de una misma operación. Aunque existen de antemano varios workflow en la instalación por defecto de Matterhorn, éstos son totalmente configurables y personalizables. El WorkflowService registrará automáticamente cualquier documento de workflow ubicado en la carpeta destinada a este fin. En la sección inicial del documento se definen los campos *id*, *title* y *description*, donde se indica el identificador del workflow, el título y una descripción del mismo. Posteriormente se indica

una lista ordenada de operaciones bajo las etiquetas *operation* que serán aplicadas al mediaPackage de manera secuencial. A continuación se muestra un pequeño ejemplo de la estructura que siguen los documentos de workflow en Matterhorn:

```
<definition>

  <!-- Descripción -->
  <id></id>
  <title></title>
  <description></description>

  <!-- Operaciones -->
  <operations>
    <operation></operation>
    ...
  </operations>

</definition>
```

Las operaciones de un workflow son normalmente llamadas a otros servicios de la plataforma. Estas operaciones son definidas dentro del servicio ConductorService de Matterhorn y son conocidas como *Workflow Operation Handler*s. Algunas operaciones básicas incluidas en Matterhorn son por ejemplo *inspect* para inspeccionar los medios y extraer datos técnicos como el formato o la duración, *compose* para la aplicación de distintos códecs de audio y vídeo o *archive* para almacenar los medios en el sistema.

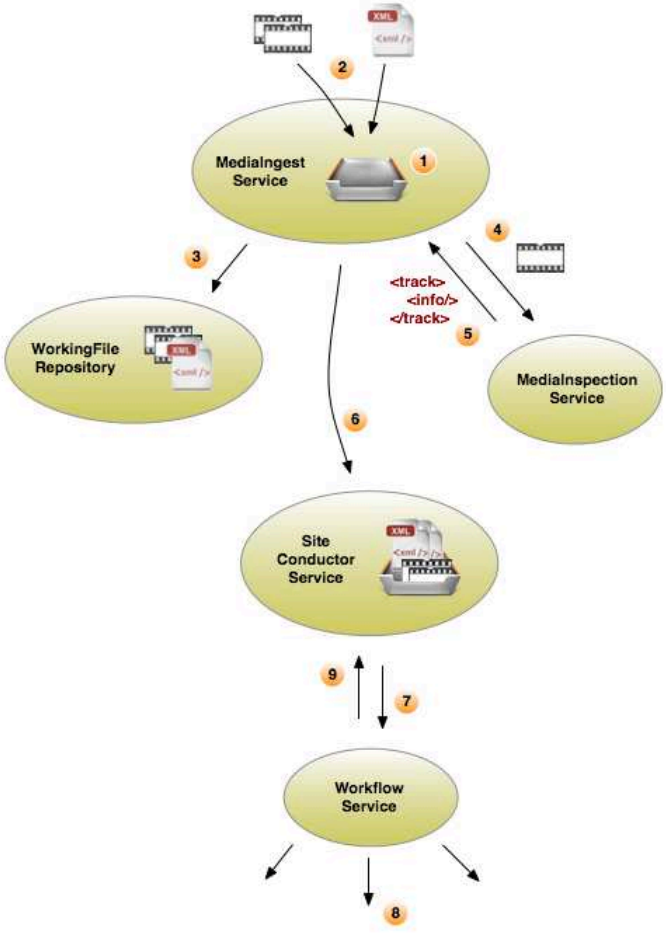

Figura 2.2: Diagrama de formación y procesamiento de un mediaPackage.

Capítulo 3

El repositorio poliMedia

Emplearemos poliMedia como repositorio de vídeos educativos sobre los que basaremos la experimentación del proyecto.

poliMedia es un sistema diseñado en la Universidad Politécnica de Valencia para la creación de contenidos multimedia de apoyo a la docencia presencial, que abarca desde la preparación del material docente hasta la distribución a través de distintos medios (TV, Internet, CD, etc.) a los destinatarios. Actualmente contiene aproximadamente 6.000 vídeos correspondientes a más de 1.000 horas de grabación.

3.1. Características y formato

poliMedia [dV12] es un sistema de producción de materiales educativos de calidad. Es un recurso integrado con todas las herramientas de PoliformaT (plataforma de teleformación de la UPV), y resulta muy adecuado como apoyo y complemento a la enseñanza presencial. Se trata de un sistema completamente innovador y único, disponible sólo en la UPV, que pone a disposición del profesorado instrumentos, materiales y técnicos para la grabación y publicación de vídeos educativos. Por ello, no requiere conocimientos audiovisuales o técnicos. El autor es el propietario intelectual de la obra.

Los vídeos principalmente consisten en la grabación de una lección por un profesor, con una duración media aproximada de 10 minutos. Los temas o asignaturas sobre las que se dispone de grabaciones son muy variados. Todas las grabaciones siguen un patrón muy similar. No han sido grabadas en distintas aulas sino que han sido realizadas en un aula habilitada y destinada exclusivamente para este fin. En todas ellas se muestra, ocupando la mayor parte de la pantalla, las transparencias o apuntes empleados por el profesor para seguir la lección. En la esquina inferior derecha, ocupando aproximadamente un cuarto de pantalla se muestra al profesor o autor del vídeo. Cabe remarcar que el área donde se muestran las diapositivas o transparencias es realmente la pantalla del ordenador que está manejando el profesor en el momento de la grabación, por lo que en ciertos vídeos no se tratará exclusivamente de diapositivas sino de cualquier tipo de ilustración que pueda realizarse desde el mismo, como

por ejemplo el manejo de un programa.

Figura 3.1: Esquema de un vídeo poliMedia

El repositorio está organizado por carpetas. Las carpetas existentes en el directorio principal del repositorio contienen los nombres ilustrativos del tema o asignatura a la que contienen. Dentro de estas carpetas existen otras subcarpetas que contienen los distintos vídeos que han sido grabados de dicho tema.

Los vídeos están almacenados en formato AVC/H.264 con diferentes configuraciones. El 85 % de ellos tienen un tamaño de 1280x720 píxeles. El bitrate[1] medio varía entre 100 y 1500 kbps, pero alrededor del 90 % presentan una tasa de bits entre 500 y 900 kbps. El formato de audio es AAC/LC estéreo (85 %) y mono (15 %) con frecuencias de muestreo de 44.100 y 48.000 Hz. En el caso del audio, el 95 % de los bitrates medios varían de 30 a 60 kbps.

3.2. Base de Datos poliMedia

Disponemos de una base de datos con información relativa a los vídeos existentes en el repositorio. Para cada uno de ellos disponemos de su identificador (consistente en un número hexadecimal de 32 caracteres generado de manera aleatoria), la ruta relativa donde está almacenado, el profesor o profesores autores del mismo, el título y el tema al que pertenece, el número de reproducciones así como otra información relevante. Una pequeña representación del contenido de la base de datos se muestra en la figura 3.2.

Haremos uso de toda esta información cuando procedamos a la construcción del reproductor prototipo basado en HTML5.

[1] Tasa de bits.

id	url	title	category	author	captions	language	view_counter
86a0b0a5-3268-8040-b732-9c424cef71cd	00240-Macromedia_Flash/M01/B01	Dibujo y Edición	Diseño Gráfico	Ivars Nicolás, Begoña	00240-Macromedia_Flash.M01.B01.trs	SP	670
c9358ca9-e4e6-0341-9d55-39c97026cc3f	00240-Macromedia_Flash/M01/B02	Organización de objetos. Capas y escenario	Diseño Gráfico	Ivars Nicolás, Begoña	00240-Macromedia_Flash.M01.B02.trs	SP	255
1bfcfda3-59d6-a848-a0ba-6f53d1bca321	00240-Macromedia_Flash/M01/B03	Animaciones simples	Diseño Gráfico	Ivars Nicolás, Begoña	00240-Macromedia_Flash.M01.B03.trs	SP	276
fe48c927-7d4b-164e-accd-86159ba104a2	00240-Macromedia_Flash/M01/B04	Animaciones complejas con Capas Guía	Diseño Gráfico	Ivars Nicolás, Begoña	00240-Macromedia_Flash.M01.B04.trs	SP	192
f3a5d582-d08c-6c43-a5cc-95d8ab06915d	00240-Macromedia_Flash/M01/B05	Clips de Película	Diseño Gráfico	Ivars Nicolás, Begoña	00240-Macromedia_Flash.M01.B05.trs	SP	154
59436990-49f6-e64d-b2de-1598776a845b	00240-Macromedia_Flash/M01/B06	Máscaras de capa	Diseño Gráfico	Ivars Nicolás, Begoña	00240-Macromedia_Flash.M01.B06.trs	SP	132
70074692-8ba8-d548-8f7a-9acf65250e29	00240-Macromedia_Flash/M01/B07	Presentación Multimedia	Diseño Gráfico	Ivars Nicolás, Begoña	00240-Macromedia_Flash.M01.B07.trs	SP	238
1d62bbd3-012f-2642-9921-9c662843cd9c	00240-Macromedia_Flash/M01/B08	Sonidos	Diseño Gráfico	Ivars Nicolás, Begoña	00240-Macromedia_Flash.M01.B08.trs	SP	107
e01a3de7-d3a4-274b-bdec-1d1f11a70d7b	00245-Antenas/M01/B01	Introducción a la agrupación de Antenas	Telecomunicaciones	Ferrando Bataller, Miguel	00245-Antenas.M01.B01.trs	SP	793
31c71a83-0be3-c142-b0d6-bfe814d1a7c6	00245-Antenas/M01/B02	Agrupaciones de dos Antenas	Telecomunicaciones	Ferrando Bataller, Miguel	00245-Antenas.M01.B02.trs	SP	328
991be4f5-eccb-6941-bef6-54ee00ed70de	00245-Antenas/M01/B03	Agrupaciones Lineales	Telecomunicaciones	Ferrando Bataller, Miguel	00245-Antenas.M01.B03.trs	SP	186
5f47b202-9ed4-cd4a-ac22-dc2454b8d396	00245-Antenas/M01/B04	Análisis de Agrupaciones - Método Gráfico	Telecomunicaciones	Ferrando Bataller, Miguel	00245-Antenas.M01.B04.trs	SP	156
22427e46-c5ec-a147-b043-08c57e7d83fd	00245-Antenas/M01/B05	Análisis de Agrupaciones - Método Gráfico II	Telecomunicaciones	Ferrando Bataller, Miguel	00245-Antenas.M01.B05.trs	SP	90
3e6c408b-045a-9a48-99cf-7692ca6e0ab8	00245-Antenas/M02/B01	Fundamentos de antenas 1ª parte: Definición de A...	Telecomunicaciones	Valero Nogueira, Alejandro		SP	433
2b624d18-85ff-cb43-8d70-a2fb8059500e	00245-Antenas/M02/B02	Fundamentos de antenas 2ª parte: Parámetros bás...	Telecomunicaciones	Valero Nogueira, Alejandro		SP	240

Figura 3.2: Muestra del contenido de la base de datos poliMedia.

3.3. Sistema de transcripción automática

Se ha construido un sistema de transcripción automática para el corpus poliMedia de forma paralela a este proyecto. Aunque el diseño e implementación de este sistema no forma parte del proyecto, supone un aspecto clave en el desarrollo del mismo pues será la base sobre la que construiremos posteriormente el sistema interactivo de edición de transcripciones. Por ello resumiremos brevemente sus características y el proceso seguido para su implementación.

El sistema de reconocimiento automático del habla se compone de una etapa de preproceso (modelado de la señal vocal), una etapa acústico-fonética (modelado acústico de unidades léxicas y/o subléxicas) y una etapa sintáctico-semántica (modelado del lenguaje). Estas dos últimas pueden ser combinadas de manera secuencial o integrarse en un único módulo.

Para posibilitar el entrenamiento de los modelos, se ha transcrito manualmente un subconjunto del corpus. En la tabla 3.1 se muestra una pequeña comparativa del volumen del subconjunto empleado para el entrenamiento respecto del corpus completo.

Cuadro 3.1: Subconjunto de poliMedia empleado

Conjunto	Horas	Videos
corpus poliMedia	1350	6830
subconjunto empleado	119	732
porcentaje total	8,85 %	10,72 %

Los parámetros de los modelos acústico y de lenguaje han de ser estimados mediante un conjunto de entrenamiento, en nuestro caso el formado por los vídeos del repositorio seleccionados y sus correspondientes transcripciones manuales. El modelo

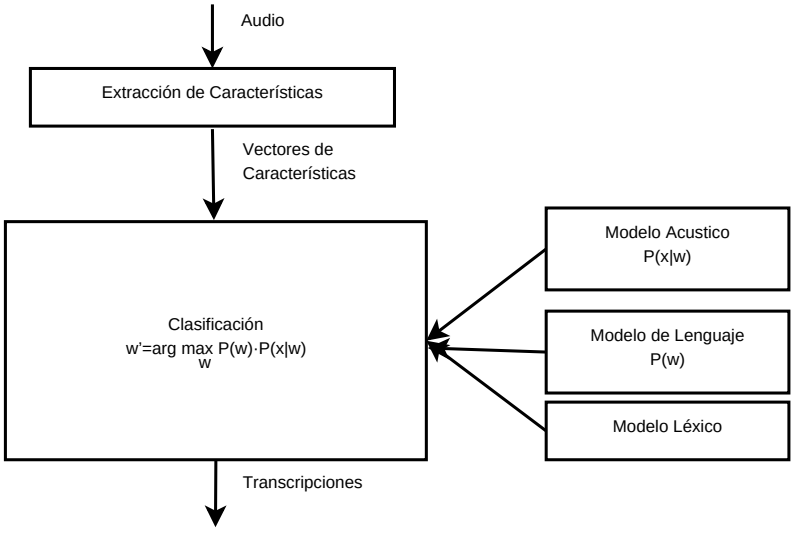

Figura 3.3: Esquema general del sistema

acústico está basado en HMMs[2] [Rab89], cuyos estados representan distintas configuraciones del aparato fonador. De una configuración se puede pasar a otra de acuerdo con ciertas reglas probabilísticas.

La figura 3.4 muestra una idea intuitiva del uso de Modelos Ocultos de Markov en el marco del reconocimiento del habla. En este caso el HMM se denomina *de izquierda a derecha* donde los estados representarían distintas configuraciones del aparato fonador. En cada estado se puede emitir un sonido de entre los de un conjunto con cierta distribución de probabilidad, que serán estimadas a partir del conjunto de entrenamiento. Las palabras se representarían mediante la concatenación de estos HMMs.

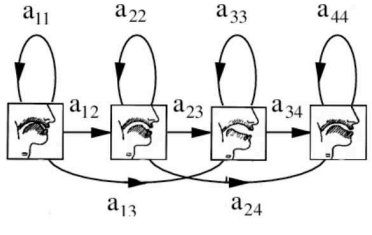

Figura 3.4: Modelos Ocultos de Markov aplicados a la señal vocal.

El bloque de modelización del lenguaje trata de aplicar las reglas gramaticales que rigen la comunicación para facilitar la comprensión de la cadena de unidades acústicas (sin contenido léxico-semántico) proporcionada por el módulo de reconocimiento

[2]Hidden Markov Models, en español: Modelos Ocultos de Markov.

acústico. El modelo de lenguaje utilizado se basa en n-gramas [Jel91], que describen la probabilidad de observar una determinada palabra dependiendo de las *n-1* anteriores. El modelo de n-gramas es calculado a partir de las transcripciones manuales disponibles. El proceso de entrenamiento se detalla gráficamente en la figura 3.5.

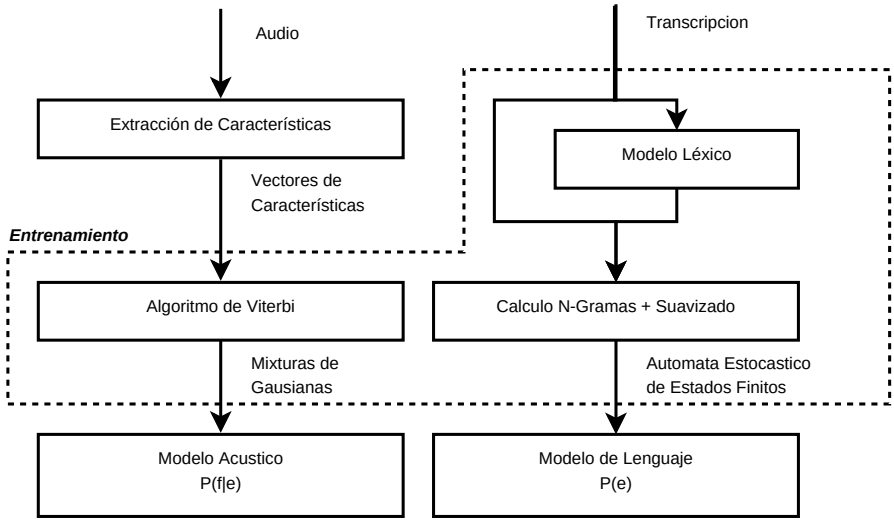

Figura 3.5: Proceso de generación de los modelos

Con el sistema finalmente implementado y optimizado, se han realizado diversas experimentaciones para determinar de manera aproximada el grado de precisión que podemos esperar del mismo. La métrica empleada para la evaluación del sistema ha sido el *Word Error Rate* (WER), y los resultados obtenidos señalan que el WER que cabe esperar del mismo está en torno a los 30-35 puntos. Esto significa que podemos esperar que en una transcripción haya un 30 %-35 % de palabras transcritas incorrectamente por término medio. Sin embargo estos resultados no son lo suficientemente precisos para ser útiles por sí mismos. Para obtener transcripciones más precisas será necesaria la intervención humana. Podemos aprovechar las transcripciones obtenidas de forma automática para construir un sistema que facilite la supervisión y corrección de éstas de un modo interactivo, y así proporcionar una solución eficiente en términos de costes y esfuerzo a la obtención de transcripciones precisas.

Capítulo 4

Reproductor prototipo HTML5

Se ha implementado un reproductor basado en el lenguaje HTML5 con el fin de proporcionar una interfaz de usuario que permita tanto el visionado como la edición interactiva de transcripciones. Este reproductor será un prototipo que nos servirá tanto para definir las características de la interfaz de edición como para diseñar el sistema de comunicación interactivo entre el editor y el sistema de reconocimiento automático del habla. Disponemos de transcripciones manuales para algunos de los vídeos del repositorio, tal y como se comentó en el apartado 3.3 cuando hablamos del entrenamiento del modelo de lenguaje. Para el resto, las transcripciones han sido obtenidas automáticamente. Como sabemos, las transcripciones obtenidas de forma automática no son suficientemente precisas y contienen errores. En este capítulo trabajaremos en el diseño de un reproductor/editor que permita al usuario, además de visualizar estas transcripciones, modificarlas de una forma interactiva. En los siguientes apartados trataremos diversos aspectos concernientes al diseño e implementación de este editor con el fin de acabar con un sistema interactivo que facilite al máximo la tarea de edición.

Para la implementación del prototipo y del resto de complementos que se indican a lo largo de este capítulo se ha empleado el sistema de gestión de bases de datos MySQL y los lenguajes de programación PHP y JavaScript junto a su popular biblioteca jQuery.

4.1. Página principal

En la página principal del reproductor se permite al usuario seleccionar entre las bases de datos disponibles para mostrar su contenido, entre las que se encuentra el repositorio poliMedia. La estructura de la página está formada por un logo superior, un menú principal, un panel lateral donde se permite seleccionar al usuario el criterio de agrupación deseado y el área de contenido. Por defecto, en el panel lateral aparecen

las distintas categorías en las que se agrupan los vídeos pertenecientes al repositorio actual. Sin embargo, mediante un panel desplegable en la parte superior del panel lateral se permite al usuario la agrupación por otros criterios, como por autor o idioma. Para la comodidad del usuario, en caso que el número de elementos en los que se agrupan los vídeos sea muy elevado, éstos serán a su vez agrupados por su letra inicial en forma de árbol desplegable.

Al seleccionar un conjunto de vídeos en el panel lateral, éstos aparecen en el área de contenido con su correspondiente información (título, autor e imagen en miniatura del vídeo) para que puedan ser seleccionados individualmente. También está disponible en el menú principal la opción de búsqueda, en la que el usuario puede definir diversos criterios de búsqueda y se le mostrarán los vídeos que concuerden con dicho criterio.

Figura 4.1: Estructura de la página principal.

4.2. Reproductor

El reproductor engloba fundamentalmente dos elementos que conviene diferenciar. Por una parte tenemos el reproductor o visor, que corresponde a la parte del reproductor donde se muestra el contenido del vídeo y al panel de controles que controla la reproducción del mismo. Por otro lado tenemos el editor, donde se mostrará la transcripción del vídeo y se permitirá al usuario realizar modificaciones. Para evitar

confusiones, de aquí en adelante las referencias al reproductor se referirán únicamente al visor.

4.2.1. Reproductor

El reproductor está basado en HTML5, por lo que inicialmente depende del navegador empleado. Para construir un reproductor personalizado que se muestre de igual manera independientemente del navegador utilizado, emplearemos hojas de estilo CSS[1] y la popular biblioteca de JavaScript jQuery.

Inicialmente se ha implementado un panel de controles con las funcionalidades básicas para la reproducción de vídeo: un botón de reproducción/pausa, un control de volumen y un posicionador temporal. Posteriormente se han añadido otros controles como un botón para mostrar/ocultar los subtítulos disponibles, un botón de selección del idioma de las transcripciones, así como los botones relativos al editor de subtítulos *Editar subtítulos* y *Transcripción imprecisa* de los que se darán más detalles en el siguiente apartado.

4.2.2. Editor

Mediante el botón *Editar subtítulos*, disponible para ciertos usuarios en el panel de controles, se muestra el editor de transcripciones. La página queda estructurada de modo que en la parte izquierda se muestra el reproductor de vídeo y en la parte derecha el editor. El editor contiene inicialmente la transcripción del vídeo, que puede ser tanto automática como manual si se dispusiera de ella. Las frases aparecen divididas del mismo modo en que se muestran los subtítulos, es decir, por códigos temporales. Para la comodidad del usuario, la reproducción del vídeo en el reproductor y el avance de las transcripciones en el editor están sincronizadas. Es decir, las transcripciones avanzan automáticamente en el reproductor a medida que el vídeo se reproduce. Además, se resalta el elemento o frase que corresponde al instante actual de reproducción para evitar confusiones.

Pulsando sobre una frase en el editor se habilita su edición. El usuario puede modificar y corregir palabras de un modo interactivo. A través de un servicio web se enviarán las correcciones del usuario al sistema de transcripción automática y éste devolverá nuevas hipótesis de transcripción a raíz de las mismas. Por ejemplo, imaginemos que durante un vídeo concreto se menciona en repetidas ocasiones un nombre propio que no es conocido por el sistema de reconocimiento automático del habla, por ejemplo "Java". Supongamos que cada vez que aparece la palabra "Java" en el vídeo, el sistema la transcribe erróneamente como "jarra". La idea será que una vez el usuario corrija la primera aparición de "jarra" por "Java", el sistema *aprenda* de dicha corrección y devuelva una nueva transcripción en la que el resto de apariciones de la palabra "Java" aparezcan transcritas correctamente. Puesto que este proceso puede suponer un alto coste computacional, distinguiremos varias modalidades de edición interactiva dependiendo del tiempo de respuesta requerido.

[1] CSS, en inglés: *Cascading Style Sheets*.

Dispondremos también de las medidas de confianza tanto a nivel global (confianza a nivel de vídeo) como para grupos de palabras o palabras sueltas, proporcionadas por el sistema de transcripción automática. Gracias a ellas podremos diseñar un modo de edición no-lineal centrado en los elementos o frases que no superen un cierto umbral de confianza (más detalle en la sección 4.4).

Por último, se han implementado algunos atajos de teclado y otros botones útiles para facilitar aún más la tarea de edición al usuario. Por ejemplo, en la frase del editor donde esté situado el cursor se mostrará un pequeño icono de reproducción que reproducirá el vídeo desde el instante en que comienza dicha frase, y servirá al usuario para agilizar las comprobaciones a nivel de frase.

Figura 4.2: Reproductor y editor de subtítulos

4.3. Tipos y roles de usuario

No todos los usuarios tienen las mismas prioridades a la hora de visualizar los contenidos educativos. Es razonable pensar, por ejemplo, que el usuario que se dispone a reproducir un vídeo para el aprendizaje de cierta materia no va a emplear mucho tiempo en la corrección de transcripciones. Por ello y por otros motivos se ha creído conveniente la distinción de los distintos usuarios en al menos dos grupos:

- **Colaborador**: Serán los usuarios que estén registrados en la base de datos como colaboradores o transcriptores profesionales. Es decir, personas que se dedican fundamentalmente a la edición y modificación de las transcripciones del sistema o que tienen acceso explícitamente para ello. Además, se incluirán en este grupo a los autores de los correspondientes vídeos. Estos usuarios dispondrán del botón "Editar transcripción" en la barra de controles del reproductor, y consecuentemente no tendrán ninguna limitación en cuanto a la edición se refiere.

- **Usuario simple**: En este grupo se encuentran el resto de usuarios del sistema. Puesto que no se espera que dediquen mucho tiempo a la edición minuciosa de las transcripciones, no disponen de acceso al editor interactivo de transcripciones. Sin embargo, disponen de otro botón en la barra de controles llamado "Transcripción imprecisa" mediante el cual los usuarios podrán reportar al sistema que la transcripción no es lo suficientemente precisa para su entendimiento.

4.4. Edición interactiva y medidas de confianza

El sistema de transcripción automática nos proporcionará medidas de confianza [FWN01]. Entendemos la medida de confianza de una cierta frase o palabra como la fiabilidad con la que el sistema cree haber reconocido correctamente dicha frase o palabra. Por tanto, una frase con un nivel de confianza *bajo* será más susceptible o tendrá más probabilidad de contener errores que otra con un alto nivel de confianza. Las medidas de confianza se estiman a nivel de palabra, y son calculadas directamente como la probabilidad a posteriori de éstas dadas todas las observaciones acústicas de la pronunciación. Las medidas de confianza de conjuntos de palabras son calculadas a partir de las medidas de confianza de las palabras individuales que conforman dicho conjunto. Esta información será de valiosa utilidad de cara a la edición interactiva: puesto que sabemos con cierta precisión qué frases o palabras de la transcripción son más susceptibles de contener errores que otras, aprovecharemos para implementar un modo de edición que, mediante algo parecido a un *potenciómetro*, permita al usuario *resaltar* dichos elementos y así agilizar el proceso de edición. Para ello será necesario emplear un formato que permita incluir medidas de confianza en las transcripciones, del que se darán más detalles en la sección 4.6.2.

Añadiremos un nuevo botón al editor que permitirá al usuario seleccionar un umbral de confianza bajo el cual se resaltarán todas aquellas palabras o frases de la transcripción cuya medida de confianza no supere dicho umbral. Tras seleccionar un cierto umbral de confianza, las frases o palabras cuya confianza sea inferior a éste serán resaltadas en el editor. Además, la edición se centrará fundamentalmente en los elementos resaltados, de modo que la reproducción del vídeo en este caso no será lineal sino que se adecuará a los intervalos de tiempo que contengan las frases o elementos resaltados.

4.5. Modos de interacción y tiempo de respuesta

En la sección 4.2 introdujimos la interacción del usuario con el sistema de transcripción, y cómo éste último a través de las ediciones del usuario devolvía nuevas hipótesis de transcripción. Aunque no entraremos en detalle en la implementación del sistema encargado de procesar las correcciones y obtener las nuevas hipótesis, sí será conveniente estudiar en profundidad la carga computacional que ello conlleva, puesto que el tiempo de respuesta que podamos obtener será crucial de cara a la interacción *usuario-máquina*. Distinguiremos tres modos de interacción:

- **No-interactivo**: En esta modalidad, el sistema no produce nuevas hipótesis. Las modificaciones hechas por el usuario son preservadas en la transcripción pero ésta no es procesada posteriormente.

- **Interactivo iterativo**: El usuario interacciona a tiempo real con el sistema de transcripción automática. Es decir, las correcciones hechas por el usuario son procesadas a *tiempo real*, con lo que al usuario se le ofrecen nuevas hipótesis conforme realiza modificaciones en la transcripción.

- **Interactivo** *batch* **o** *journal*: En esta última modalidad de interacción, el usuario realiza modificaciones sin que el sistema le ofrezca nuevas hipótesis a raíz de éstas. Sin embargo, a diferencia del modo *no-interactivo*, estas modificaciones serán procesadas con posterioridad y la transcripción que prevalecerá finalmente será la obtenida por el sistema de transcripción automática tras procesar la transcripción modificada por el usuario.

La comunicación entre la interfaz de usuario (el editor) y el sistema de transcripción automática se implementará mediante un servicio web encargado de transmitir la información necesaria en ambos sentidos: las modificaciones realizadas por el usuario al sistema de transcripción automática y las nuevas transcripciones al reproductor/editor de transcripciones (ver figura 4.3).

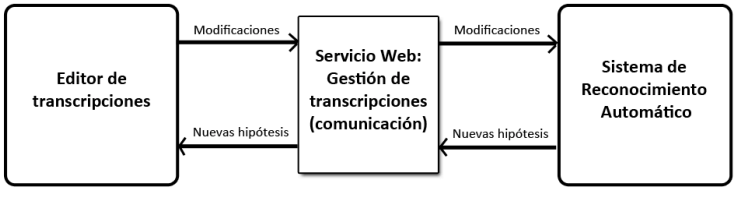

Figura 4.3: Servicio web para la gestión de transcripciones.

4.6. Formatos de vídeo y transcripciones

4.6.1. Formatos de vídeo en HTML5

Como vimos en el capítulo 3, los vídeos están almacenados originalmente en formato MPEG-4 y tienen un tamaño de 1280x720 píxels. El formato del audio es AAC (Advanced Audio Codec) a 44.1kHz, estéreo. Sin embargo, nos encontramos con ciertos problemas de compatibilidad para algunos navegadores. Puesto que la adaptación de los principales navegadores a HTML5 se encuentra, a fecha de la publicación de este proyecto, aún en fases tempranas de desarrollo y todavía no se han adaptado definitivamente a los nuevos elementos multimedia, algunos de ellos no soportan la reproducción de vídeo en formato MPEG-4, como por ejemplo *Mozilla Firefox*. Por suerte, el *tag* <video>de HTML5 soporta la inclusión de distintas fuentes de vídeo (en distintos formatos) para la reproducción, como se muestra en el siguiente ejemplo:

```
<video width="640" height="480" controls="controls">
   <source src="video.mp4" type="video/mp4" />
   <source src="video.ogg" type="video/ogg" />
</video>
```

Por este motivo se ha replicado el repositorio original aplicando el códec Theora[2], soportado también por *Firefox* y la mayoría de navegadores. Con ambos formatos disponibles, aseguramos la compatibilidad con la gran mayoría de navegadores actuales.

4.6.2. Formato de transcripciones

En cuanto al formato que seguirán las transcripciones en el sistema, necesitaremos un formato flexible que permita la extensión del mismo con nuevas etiquetas para la representación de información relevante en la transcripción como las medidas de confianza (sección 4.4), qué ha sido modificado manualmente en una transcripción, posibilitar la representación de dos transcripciones alternativas para una misma frase/palabra (sobretodo interesante de cara a la traducción), etc. Por ello se ha decidido proponer una extensión para el lenguaje TTML/DFXP [TTM10].

El formato DFXP[3] [Gö10] está basado en el lenguaje TTML[4]. Se trata de un formato XML que da soporte al uso de subtítulos para reproductores y aplicaciones web en general. DFXP ofrece una serie de herramientas creativas que permiten entre otras cosas, determinar la situación de los subtítulos en la pantalla, el uso de diversas fuentes, incluir metadatos o algunas posibilidades para la representación de texto animado. A continuación se adjunta una pequeña muestra del formato básico que sigue el contenido de una transcripción en un documento DFXP:

...

[2] http://www.theora.org
[3] Distribution Format Exchange Profile.
[4] Timed Text Markup Language.

```
<body region="subtitleArea">
  <div>
    <p xml:id="subtitle1" begin="0.76s" end="3.45s">
      Hola, ¿qué tal va todo?
    </p>
    <p xml:id="subtitle2" begin="5.0s" end="10.0s">
      Estupendamente,<br/>
      ¿qué haces por aquí?
    </p>
...
```

Necesitaremos añadir nuevas etiquetas para incluir nueva información en las transcripciones. Las nuevas etiquetas propuestas son las siguientes:

- <#globalconfidence value="XX"/>
  Esta etiqueta servirá para añadir la confianza del vídeo a nivel global. Se incluirá en la cabecera del documento DFXP y aparecerá una única vez.

- <#conf value="XX"></#conf>
  Indicará que las palabras situadas en el interior de esta etiqueta tienen un valor de confianza "XX". Puede incluir tanto palabras sueltas, como grupos de palabras o párrafos completos.

- <#alternate><#altoption></#altoption></#alternate>
  Esta etiqueta será especialmente útil de cara a las traducciones. Permite representar diferentes posibilidades de transcripción para una misma frase o palabra. Un ejemplo de uso sería el siguiente:

  ```
  Hola, ¿
  <#alternate>
    <#altoption>cómo estás</#altoption>
    <#altoption>cómo vas</#altoption>
  </#alternate>
  ?
  ```

- <#manual author="id"></#manual>
  Las palabras contenidas por esta etiqueta corresponderán a palabras que han sido editadas o añadidas manualmente por una persona, y nunca por el sistema de transcripción automática. Se almacenará quién ha realizado la modificación en el campo *author*.

Las etiquetas pueden anidarse conforme sea necesario. Un ejemplo de uso de varias de las nuevas etiquetas para un cierto párrafo sería el siguiente:

```
Hola, ¿
<#alternate>
```

```
  <#altoption><#manual author="Alex">cómo estás</#manual></#altoption>
  <#altoption><#conf value="0.5">cómo atrás</#conf></#altoption>
</#alternate>
?
```

El ejemplo anterior contemplaría el caso en el que inicialmente el sistema de transcripción automática hubiese propuesto como transcripción para un cierto intervalo de tiempo t la frase "Hola, ¿cómo atrás?", siendo la confianza del grupo de palabras "cómo atrás" igual a 0.5. Posteriormente, el usuario "Alex" habría editado manualmente dicha transcripción y habría indicado que la transcripción correcta para "cómo atrás" es "cómo estás".

Capítulo 5

Integración en la plataforma Matterhorn

5.1. Perspectiva

El objetivo de este capítulo es integrar en la plataforma Matterhorn las técnicas de transcripción asistida que fueron definidas a lo largo del capítulo 4 para el reproductor prototipo basado en HTML5. Existen diferentes posibilidades de cara a llevar a cabo esta integración, dependiendo principalmente de la distribución del sistema de transcripciones y de las necesidades del modelo. Una posibilidad sería integrar el sistema de transcripción automática en el mismo sistema Matterhorn, y atribuir a Matterhorn la gestión de las transcripciones. Sin embargo, la propuesta seguida en los sucesivos apartados de este capítulo sugiere un sistema distribuido donde el sistema de transcripción automática se encarga por separado de la gestión de las mismas (obtención, almacenamiento, etc.). La integración en Matterhorn constará por tanto de dos aspectos fundamentales: el primero la necesidad de transmitir al sistema de transcripción automática los nuevos contenidos audiovisuales añadidos al sistema, y el segundo la adaptación de un nuevo reproductor para la plataforma Matterhorn que permita la visualización y edición interactiva de transcripciones. Ambas dos tareas incluirán la extensión y adaptación del servicio web empleado en el reproductor prototipo (ver fig. 4.3) para cubrir las necesidades de comunicación entre la plataforma Matterhorn y el sistema de transcripción automática (a partir de ahora Sistema de Gestión de Transcripciones). De este servicio web se darán más detalles en la siguiente sección.

5.2. Sistema de Gestión de Transcripciones e interfaces

Deberán definirse primeramente una serie de interfaces para permitir la comunicación con el Sistema de Gestión de Transcripciones. La idea es basar todas estas interfaces en llamadas HTTP REST[1] [Tya06]. Una de las caraterísticas clave de los servicios web REST es el uso explícito de los métodos HTTP. Estamos hablando, por tanto, de crear un servicio web tipo REST para gestionar el intercambio de información entre Matterhorn y el Sistema de Gestión de Transcripciones. Necesitaremos definir varias interfaces, por ejemplo para la subida de archivos al sistema, para conocer el estado de procesamiento de los archivos enviados o para obtener las transcripciones de un vídeo determinado. A continuación se describen con más detalle las principales interfaces implementadas:

- **Ingest**: Será un servicio POST mediante el cual se añadirá un nuevo elemento al Sistema de Gestión de Transcripciones para ser procesado. El identificador "id" deberá coincidir con el identificador del *mediaPackage* creado en Matterhorn para la posible identificación a posteriori.

Método /Ruta	POST /ingest
Descripción	Añade un nuevo elemento (vídeo/audio) al sistema
Parámetros requeridos	id: Identificador del elemento (mediaPackage ID de Matterhorn)
Body (subida)	El archivo multimedia

- **Status**: Empleado para conocer el estado de procesamiento de un elemento del sistema no procesado, procesando, transcrito.

Método /Ruta	GET /status
Descripción	Devuelve el estado de procesamiento del elemento *id*.
Parámetros requeridos	id: Identificador del elemento (mediaPackage ID de Matterhorn)
Formato de respuesta	text/xml

- **DFXP**: Devolverá la transcripción de un vídeo concreto del sistema en formato DFXP en caso de disponer de ella.

[1] REST: REpresentational State Transfer.

Método /Ruta	GET /dfxp
Descripción	Devuelve la transcripción del elemento *id* en formato DFXP.
Parámetros requeridos	id: Identificador del elemento (mediaPackage ID de Matterhorn)
Formato de respuesta	text/xml

- **Mod**: Posibilitará el envío de correcciones o modificaciones para la transcripción de un elemento del sistema, que deberán seguir un formato determinado.

Método /Ruta	POST /mod
Descripción	Envío de correcciones para la transcripción del elemento *id*.
Parámetros requeridos	id: Identificador del elemento (mediaPackage ID de Matterhorn)
Body (subida)	Modificaciones realizadas (formato concreto).

5.3. Parte I: Integración en el workflow de Matterhorn

Ya están definidas e implementadas las interfaces para la comunicación con el Sistema de Gestión de Transcripciones. Tenemos la necesidad de transmitir los nuevos medios añadidos a Matterhorn al SGT (Sistema de Gestión de Transcripciones) para que éstos sean procesados. Como se comentó en la sección 5.1, de la gestión de las transcripciones se encargará por completo el SGT. Esto implica que las transcripciones se almacenarán también en el SGT y no en los *mediaPackages* del sistema Matterhorn. En Matterhorn, cada elemento añadido al sistema es procesado mediante una serie de operaciones que conforman un workflow (ver secciones 2.2.1 y 2.2.2). El procedimiento será crear un nuevo servicio encargado de transmitir la información al SGT y la correspondiente *workflow operation* que haga uso de éste, para posteriormente añadirla al workflow por defecto de Matterhorn. La figura 5.1 ilustra el proceso que seguiría un nuevo vídeo añadido al sistema.

A continuación se describen los pasos seguidos para la creación de este nuevo servicio y su posterior implantación en el sistema:

1. **Implementar el nuevo servicio.**

   Primero se creará la estructura de archivos y directorios del servicio. El código del nuevo servicio estará ubicado en dos carpetas en el directorio *modules* dentro del directorio base de Matterhorn, bajo los nombres matterhorn-transcription-service-api y matterhorn-transcription-service-impl. En el primero de ellos sim-

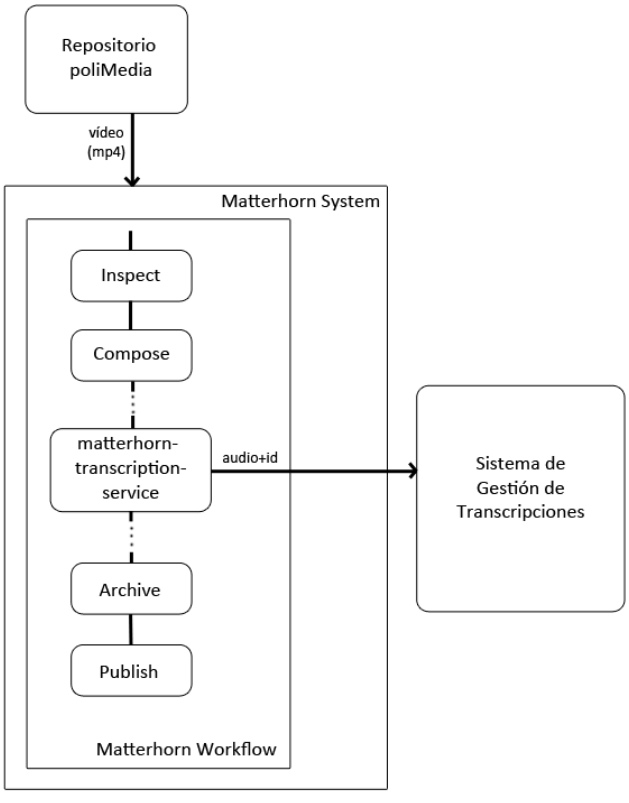

Figura 5.1: Procesamiento de un vídeo a través del workflow de Matterhorn

plemente se implementará la API[2] de Java del servicio. En el directorio ...-service-impl se implementará dicha interfaz y sus REST Endpoints. Aunque el resto de directorios y archivos puede generarse manualmente, es recomendable partir de otro servicio ya implementado y realizar las modificaciones que sean pertinentes. Este servicio transmitirá las pistas de audio y vídeo al Sistema de Gestión de Transcripciones mediante la llamada REST *ingest* del servicio web definido en la sección 5.2.

2. Crear un Workflow Operation Handler.

El siguiente paso es la implementación del controlador de operación que hace uso del nuevo servicio. Las operaciones de workflow se definen dentro del módulo matterhorn-conductor. Para ello se crean dos nuevos archivos:

- un archivo Java que contiene la implementación de la operación, llama-

[2]Application Programming Interface, en español: Interfaz de programación de aplicaciones.

do TranscriptionWorkflowOperationHandler.java y situado en matterhorn-conductor/src/main/java junto al resto de controladores de operación.

- un archivo XML que contiene la declaración OSGi, situado en matterhorn-conductor/src/main/resources/OSGI-INF/operations.

Por último se declara la operación en el archivo pom.xml del matterhorn-conductor.

3. **Incluir la operación en el workflow.**

   Una vez implementado el controlador de operación para el nuevo servicio, ya podremos ejecutar la operación desde cualquier workflow. Se puede elegir si definir un nuevo workflow que haga uso de ésta o simplemente modificar el workflow por defecto de Matterhorn (compose-distribute-publish.xml) para que incluya la nueva operación.

5.4. Parte II: Sustitución/adaptación del Engage Media Player

El Engage Media Player es el reproductor de medios de la plataforma Matterhorn. La tecnología en la que se basa es puramente HTML/JavaScript junto a un contenedor de vídeo basado en Flash 10. Entre sus características se encuentran, además de la mera reproducción de los contenidos audiovisuales: la visualización de subtítulos, la reproducción de vídeo selectiva correspondiendo a las distintas diapositivas de la presentación, la búsqueda basada en contenido, estadísticas de visualización o anotaciones de los usuarios. Sin embargo, en nuestro caso necesitaremos un reproductor que disponga además de funcionalidad para la edición y modificación de transcripciones, como vimos en el capítulo 4. Cabe mencionar aquí el esfuerzo que el Área de Sistemas de Información y Comunicaciones (ASIC) de la UPV está haciendo en la construcción de un reproductor alternativo basado en HTML5 compatible con Matterhorn bajo el nombre *Paella Engage Player* [UPV12]. El proyecto se encuentra todavía en fase de desarrollo a fecha de la publicación de este proyecto, y si bien todavía no dispone de funcionalidad para la edición de transcripciones, ésta se encuentra entre sus posibles objetivos a corto plazo. La figura 5.2 muestra una versión extendida del Mini-Paella Player (una versión idéntica al Matterhorn Engage Player oficial pero basada en HTML5) que posibilita la edición de transcripciones.

Dado el modelo distribuido de los sistemas propuesto en la sección 5.1, necesitaremos que el reproductor implementado se comunique con el Sistema de Gestión de Transcripciones a través del servicio REST definido en la sección 5.2, tanto para la obtención de las transcripciones como para el intercambio interactivo de modificaciones. Esta comunicación se ilustra en la figura 5.3.

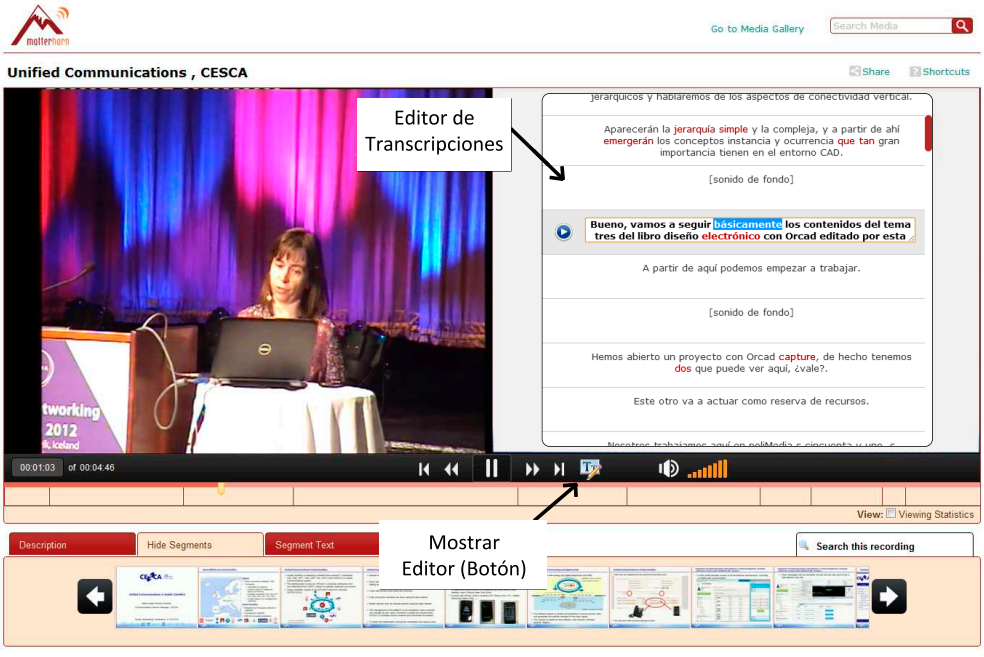

Figura 5.2: Matterhorn Engage Player con edición de transcripciones.

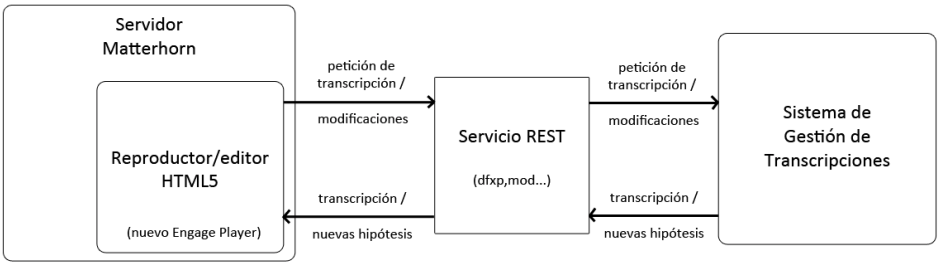

Figura 5.3: Comunicación entre el reproductor HTML5 y el SGT.

Capítulo 6

Conclusiones y trabajos futuros

6.1. Resumen

A lo largo de este trabajo hemos señalado la necesidad de disponer de una solución para la obtención de transcripciones de calidad de un modo eficiente. La propuesta ha consistido en unificar un sistema automático de reconocimiento del habla con un sistema de edición interactivo que permita la supervisión de estas transcripciones automáticas de una manera eficaz. Para la construcción de este sistema complejo hemos dispuesto de un conjunto de datos suficientemente amplio y de una plataforma de pruebas que nos ha permitido experimentar las necesidades reales de los usuarios. En el capítulo 1 hemos introducido la problemática del reconocimiento automático del habla. Hemos dado un breve repaso a la historia reciente de estos sistemas, y hemos visto cómo han ido avanzado sus prestaciones hasta la actualidad. La falta de precisión de estos sistemas nos ha servido para justificar la interacción con el usuario en el proceso de transcripción, que es uno de los objetivos principales de este trabajo. A continuación, en el capítulo 2 hemos introducido al lector la plataforma Matterhorn, dando una visión general de las características de la misma y detallando algunos aspectos que se han considerado de relevancia para capítulos posteriores. En el capítulo 3 hemos presentado el repositorio poliMedia, sus características principales y sus contenidos. Además, hemos resumido brevemente el diseño e implementación de un sistema de reconocimiento automático del habla para este corpus. Posteriormente, en el capítulo 4 hemos visto con detalle la implementación de un reproductor prototipo basado en HTML5 para el repositorio poliMedia. Hemos dotado a este prototipo de una interfaz de usuario para la edición y modificación de transcripciones, presentando diversas técnicas e ideas para hacer esta edición lo más interactiva y eficiente posible. Por último, en el capítulo 5 hemos propuesto un modelo de integración de un sistema de transcripción semi-asistida con las mismas funcionalidades que el reproductor prototipo del capítulo anterior en la plataforma Matterhorn, y hemos visto con cierto

detalle la manera de llevar a cabo esta integración.

6.2. Conclusiones

En el capítulo 3 hemos visto como efectivamente, las transcripciones obtenidas de manera puramente automática no son lo suficientemente precisas (secc. 3.3). Para obtener transcripciones que presenten un grado de precisión óptimo en la actualidad es necesaria la intervención humana. Hemos podido comprobar cómo, partiendo de transcripciones automáticas lo suficientemente precisas (en torno al 70 % de palabras correctamente transcritas), el uso de un sistema interactivo que asista al usuario en el proceso de corrección de éstas hace mucho más eficiente la tarea. Las funcionalidades diseñadas para la edición interactiva se basan en la experiencia de diversas personas experimentadas en el campo de la transcripción y traducción manuales.

6.3. Trabajos futuros

Disponemos finalmente de un sistema eficiente de obtención de transcripciones precisas para repositorios de gran tamaño y amplio vocabulario. Puesto que el reconocimiento automático del habla sigue siendo hoy en día uno de los principales focos de investigación en el área del reconocimiento de formas, habrá que tener en cuenta las aportaciones y mejoras que surjan a lo largo de los años para mejorar de forma continua la obtención automática de transcripciones. Por tanto, este proyecto no consiste en un trabajo *con punto y final* sino que representa una base sólida sobre la que implementar futuras mejoras que hagan todavía más eficiente la obtención de transcripciones precisas. Lo mismo puede decirse sobre las técnicas interactivas de edición, que pueden ser extendidas conforme sea necesario para facilitar al usuario la modificación manual de las transcripciones. Estas funcionalidades han sido integradas en la plataforma Opencast Matterhorn con el fin de obtener no solamente un sistema prototipo que sea empleado casi exclusivamente en un laboratorio a modo de prueba de concepto, sino un sistema que permita su uso en un entorno real para poder evaluar así las necesidades reales de los usuarios.

Pero no nos olvidemos del fin último del que este proyecto representa únicamente un eslabón: facilitar el visionado de los contenidos al mayor número de personas posible. Tal y como comentamos en el capítulo 1, esto incluye la traducción de estas transcripciones a otras lenguas. La obtención de transcripciones precisas es el paso previo a la traducción automática, o en cualquier caso al proceso de traducción en general. Habrá que centrarse por tanto en la ampliación de este sistema para cubrir la diversidad lingüística de los usuarios.

Este trabajo y el posterior trabajo futuro se desarollará en el marco del proyecto europeo *transLectures*[1]. transLectures es un proyecto STReP[2] dentro del Seventh Framework Programme fundado por la Comisión Europea. Comenzó el 1 de Noviembre de 2011 y su fecha de finalización está programada para el 31 de Octubre de

[1] http://translectures.eu
[2] Specific Targeted Research Projects.

2014. El objetivo de transLectures es desarrollar soluciones novedosas y eficientes para la producción de transcripciones y traducciones precisas para el portal VideoLectures.NET, y de forma más general para otros repositorios relacionados con la plataforma Matterhorn. De este modo, los esfuerzos se centrarán en aunar un sistema de transcripción interactivo de características similares al descrito en este trabajo con un sistema de traducción que permita además disponer de estas transcripciones en diversos idiomas, y que a su vez sea compatible con la plataforma Matterhorn para poder poner a disposición de las entidades que emplean esta plataforma las soluciones alcanzadas durante el transcurso del proyecto.

Bibliografía

[Apa] Apache. Apache felix. http://felix.apache.org/site/index.html.

[dV12] Universidad Politécnica de Valencia. Web oficial del repositorio de polimedia. http://polimedia.blogs.upv.es/, 2012.

[FWN01] Klaus Macherey Frank Wessel, Ralf Schlüter and Hermann Ney. Confidence measures for large vocabulary continuous speech recognition. *IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING*, 9(3), March 2001.

[Gö10] Larissa Görner. Dfxp – subtitling in xml and mxf, 2010.

[Jel91] F. Jelinek. Up from trigrams: the struggle for improved language models. In *Proc. of the Eurospeech 91*, pages 1037–1039, September 1991.

[Jua98] B. H. Juang. The past, present, and future of speech processing. *IEEE Signal Processing Magazine*, pages 24–48, May 1998.

[KHDB52] R. Biddulph K. H. Davis and S. Balashek. Automatic recognition of spoken digits. *Acoustic Society of America*, 24(6):637–642, November 1952.

[Lee89] K. F. Lee. *Automatic Speech Recognition: The Development of the SPHINX System*. Kluwer Academic Publishers, 1989.

[LRRL96] B. H. Juang L. R. Rabiner and C. H. Lee. *An Overview of Automatic Speech Recognition*. C. H. Lee, F. K. Soong and K. K. Paliwal editores, Kluwer Academic Publisher, 1996.

[Rab89] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proc. of the IEEE*, volume 77, pages 257–286, February 1989.

[RJ86] L. R. Rabiner and B. H. Juang. An introduction to hidden markov models. *IEEE ASSP Magazine*, 3(1):4–16, January 1986.

[RJ93] L. R. Rabiner and B. H. Juang. *Fundamentals of Speech Recognition*. Prentice-Hall, Englewood Cliffs, N. J., 1993.

[SH07] Richard S. Hall. Osgi r4 service platform: Java modularity and beyond, March 2007.

[TTM10] Timed text markup language (ttml) 1.0. http://www.w3.org/TR/ttaf1-dfxp, 2010.

[Tya06] Sameer Tyagi. Restful web services. http://www.oracle.com/technetwork/articles/javase/index-137171.html, 2006.

[UPV12] UPV. Paella engage player: A multistream video player compatible with matterhorn. http://paellaengage.webs.upv.es, 2012.