Universidad
Politécnica
de Cartagena

Estudio para la clasificación parcialmente supervisada de emociones a partir de voz, utilizando métodos de autoaprendizaje y SVMs.

Proyecto Final de Carrera

Escuela Técnica Superior de Ingeniería de Telecomunicación

Universidad Politécnica de Cartagena

presentado por: José Domingo Esparza García

2011

Supervisor:	Dr. Jorge Larrey
Supervisor:	Dr. Stefan Scherer
Día de entrega:	Tag. Monat 2011

Resumen

Las interacciones hombre-máquina hacen uso de muchos canales de comunicación diferentes con el fin de obtener buenos resultados, en términos de comprensión. Los canales de audio y vídeo son los más informativos para los humanos, debido a la gran cantidad de información disponible en ellos. Para las máquinas, sin embargo, existen muchas variables que dependen del contexto que hacen la tarea mucho más complicada, tal y como pueden ser las emociones del hablante en cada instante. Las emociones pueden afectar en gran medida las expresiones y deben, por tanto, ser consideradas como tal. El trabajo realizado para este proyecto tiene como finalidad el estudio de la clasificación automática de emociones a partir de voz, utilizando clasificadores del tipo máquinas de vectores de soporte difusas.

Al mismo tiempo, las soluciones tradicionales de clasificación automática dependen de grandes conjuntos de entrenamiento que permiten aprender los patrones que hay detrás de la distribución generatriz de los datos. Puesto que el etiquetado de los datos puede suponer una tarea muy costosa tanto en tiempo como en dinero, distintas técnicas han sido propuestas en la literatura para reducir dichos costes. Este estudio también incorpora una evaluación de diferentes métodos de entrenamiento parcialmente supervisado. Las conclusiones obtenidas al respecto podrán ser extendidas a cualquier otro problema de clasificación, considerando que se trata de procedimientos estándar, aplicables a cualquier campo de estudio.

Referencias a los textos y publicaciones originales son proporcionadas en cada sección de este proyecto. Cada una de ellas es recomendada para el lector en caso de que se desee un conocimiento más detallado y en profundidad de los métodos aquí descritos.

Índice general

1 Introducción

La clasificación de emociones es realizada por humanos en todo tipo de situaciones pero, incluso para nosotros, no es siempre una tarea fácil. La literatura muestra que los tests de percepción realizados en humanos no siempre producen resultados libres de errores (Wendt, 2007). Al contrario, existen patrones de confusión entre emociones que pueden ser observados con una cierta frecuencia. Por tanto, es crucial encontrar conjuntos de rasgos y técnicas de clasificación que sean comparables a la percepción humana. De otro modo, podría ocurrir que la máquina reconociera expresiones basándose en artefactos y no en la modulación real causada por el estado emocional humano.

En este estudio, nuestra intención es emular las capacidades perceptivas humanas para demostrar que mediante una elección adecuada de rasgos y entrenamiento exhaustivo, precisiones y confusiones similares a las de los humanos pueden ser obtenidas con conjuntos de entrenamiento suficientemente grandes. Los experimentos están basados en una máquina de vectores de soporte (SVM) combinando ocho tipos distintos de rasgos para conjuntos de datos estándar. El uso conjunto de distintos rasgos requiere de una cierta fusión entre ellos que optimice la cantidad de información que pueden aportar. Una sencilla técnica de fusión es propuesta y utilizada en nuestros experimentos, proporcionando buenos resultados. El proceso de entrenamiento, sin embargo, implica un trabajo tedioso de etiquetado mediante la opinión de un experto. Este trabajo puede ser, en general, muy costoso. No obstante, la obtención de muestras sin etiqueta no incurre necesariamente un elevado coste. Por esta razón, existen lineas de investigación enfocadas en utilizar muestras no etiquetadas para entrenar el sistema. Con esta idea, pueden encontrarse distintas técnicas, cada una de ellas enfocada a aprovechar distintas características del proceso de entrenamiento. Existen estudios previos, por ejemplo, sobre aprendizaje semi-supervisado, para el cuál se utilizan muestras tanto etiquetadas como etiquetadas para entrenar (Druck et al., 2008; Zhu, 2005; Blum and Mitchell, 1998), entrenamiento no supervisado, donde solamente muestras sin etiqueta son utilizadas (eg. Clustering algorithms - Duda et al. (2001)) o aprendizaje activo, en el cuál se permite al sistema elegir su propio conjunto de entrenamiento de entre un conjunto de muestras disponibles (Lomasky et al., 2007).

Dentro del marco del aprendizaje parcialmente supervisado, en este trabajo estudiaremos una aproximación semi-supervisada, basada en el algoritmo k-nearest neighbors, así como entrenamiento activo. Para poder analizar correctamente la bondad de los métodos propuestos, una medida de confianza para las etiquetas generadas artificialmente será propuesta. De este modo será posible obtener una medida de cuán correcta es una etiqueta artificial o cómo de representativa es una muestra, según el caso de estudio.

En lo que concerne a aplicaciones reales para este estudio, existen distintas situaciones en las que el reconocimiento automático de emociones podría ser de gran ayuda. Podría plantearse, por ejemplo, un feedback automático para servicios ofrecidos mediante un operador telefónico o, más aún, adaptación de modelos para una interacción hombre-máquina particular, dependiendo del estado de ánimo. Aún más interesantes tal vez pueden ser los resultados obtenidos en los experimentos de aprendizaje parcialmente supervisado. Las conclusiones aquí obtenidas pueden ser extendidas a cualquier problema de clasificación que requiera de un proceso de entrenamiento, proporcionando soluciones que permiten reducir la cantidad de etiquetas requerida y, por tanto, el coste del sistema.

Por tanto, los objetivos de este proyecto pueden ser sintetizados como:

- Encontrar una buena combinación de rasgos que represente las capacidades perceptivas humanas.
- Desarrollar, si necesario, nuevos conjuntos de rasgos que mejoren las confusiones entre clases.
- Crear un clasificador multi-clase para la tarea de reconocimiento de emociones.
- Definir una fusión que combine las distintas decisiones obtenidas por separado en el dominio de cada conjunto de rasgos.
- Estudiar técnicas de amprendizaje parcialmente supervisado dentro de la arquitectura del sistema propuesto. Study partially supervised techniques within the proposed system architecture.

La estructura de este proyecto está organizada del siguiente modo: el Capítulo 2 introduce una descripción de los conjuntos de datos utilizados en este estudio, el Capítulo 3 proporciona una base a nivel teórico y de implementación de los métodos y técnicas utilizadas. La configuración experimental es descrita en el Capítulo 4. Una discusión sobre los resultados obtenidos y su comparación con los resultados producidos por humanos se incluye en la Sección 4.3. El Capítulo 5 presenta una síntesis del trabajo realizado y resultados obtenidos, concluyendo este proyecto.

2 Conjuntos de datos

En los comienzos de esta estudio, surgieron distintas opiniones sobre el tipo de datos a utilizar. La clasificación de emociones podría englobar el uso de distintos tipos de datos, tales como voz, vídeo o medidas biométricas (Keltner and Ekman, 2003; Keltner et al., 2003). Estudios anteriores demostraron que con información de la voz únicamente se pueden obtener buenos resultados sin incurrir en complicados procesos de sincronización. Por tanto, este estudio se limita únicamente a este tipo de datos. Los dos conjuntos de datos utlizados son descritos en este capítulo. Ambos están constituídos por una serie de emociones de referencia, entre las cuales se encuentran la más representativas, con una buena calidad de audio. Sin embargo, se trata de emociones actuadas, por lo que se esperan diferencias con respecto a situaciones reales.

2.1. WaSeP Corpus

El principal conjunto de datos utilizado para este estudio es el "Corpus of spoken words for studies of auditory speech and emotional prosody processing"(WaSeP©) Wendt and Scheich (2002). WaSeP está basado en el idioma Alemán y está estructurado del siguiente modo: una primera parte del conjunto contiene sustantivos estándar alemanes. Una segunda parte contiene pseudo palabras fonéticamente equilibradas que se ajustan a las reglas fonéticas alemanas. Únicamente el subconjunto con pseudo palabras ha sido utilizado en este trabajo, estando formado por 222 palabras, interprestadas por un hombre y una mujer, imitando 6 emociones naturales humanas: neutral, alegría, tristeza, enojo, miedo y repugnancia. Las voces no fueron grabadas en entornos realmente emocionales sino en un estilo actuado en una cámara acústica. Para validar las emociones actuadas, se pidió a un grupo de 74 hablantes nativos alemanes que escucharan las grabaciones y emitieran una decisión para cada una de las muestras, basándose en su percepción. Una porcentaje de acierto medio del 78.53 % fue alcanzado en este caso. Los resultados de estos tests pueden observarse en el Cuadro 2.1 en forma de matriz de confusión. Esta matriz proporciona una buena representación de los aciertos obtenidos por las personas reconociendo emociones, así como de las confusiones más comunes entre las distintas emociones. El mismo tipo de matrices serán obtenidas para

presentar los resultados de los experimentos y permitir una fácil comparación. El conjunto de datos original fue muestreado a 44.1 kHz y cuantificado con 16 bits. Para nuestros experimentos, los datos fueron remuestreados a 16 kHz. Un ejemplo de la señal de audio y su correspondiente espectrograma se muestran en las Figuras 2.1 y 2.2, respectivamente.

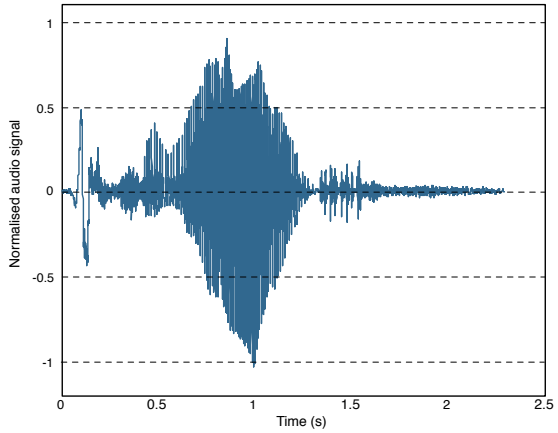


Figura 2.1: Ejemplo de señal de audio utilizada del corpus WaSeP, normalizada y remuestreada a 16 kHz.

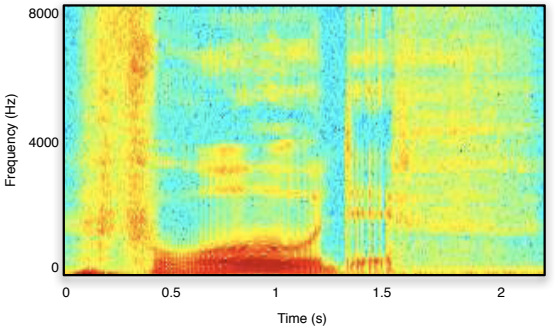


Figura 2.2: Ejemplo de señal de audio utilizada del corpus WaSeP. Espectrograma de la señal remuestreada a 16 kHz.

2.2. Berlin Database of Emotional Speech (EmoDB)

Para comparar los resultados obtenidos con un conjunto de datos más ampliamente conocido, "Database of German Emotional Speech (EmoDB)" ha sido utilizado como un conjunto de referencia en este estudio. Este conjunto de

Cuadro 2.1: Matriz de confusión para los tests conducidos para la percepción humana, generada a partir de las etiquetas disponibles para cada una de las grabacioens listadas en el conjunto WaSeP. Wendt (2007).

	F	D	H	N	S	A
Fear	.77	.01	.08	.03	.10	.01
Disgust	.05	.72	.06	.03	.07	.07
Happiness	.01	.00	.75	.22	.02	.00
Neutral	.01	.02	.05	.79	.00	.13
Sadness	.05	.01	.04	.13	.76	.01
Anger	.01	.03	.00	.01	.01	.94

datos de voz incluye grabaciones de 10 actores, tanto hombres como mujeres. Frases cortas y largas son utilizadas, representando 7 tipos de emociones diferentes (las mismas que WaSeP más aburrimiento). EmoDB fue grabada en una cámara anecoica en la Technische Universität Berlin, en el departamento de Technical Acoustics. El audio fue grabado con una frecuencia de muestreo de 48 kHz usando un micrófono Sennheiser MKH40 P48 y una grabadora Tascam DA-P1 (Burkhardt et al., 2005). Más tarde, las muestras fueron remuestreadas a 16 kHz para este estudio. Este conjunto de datos ha constituido la base de muchos estudios Scherer et al. (2008, 2007); Vlasenko et al. (2007); Wagner et al. (2007). Un ejemplo de señal de voz con su espectrograma correspondiente se puede observar en las Figuras 2.3 y 2.4.

De manera similar al conjunto WaSeP, tests de percepción humana fueron realizados con EmoDB y la matriz de confusión se muestra en el Cuadro 2.2.

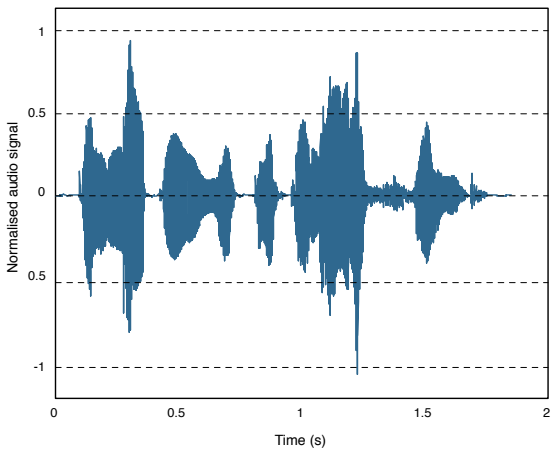


Figura 2.3: Ejemplo de señal de audio utilizada del corpus EmoDB, normalizada y remuestreada a 16 kHz.

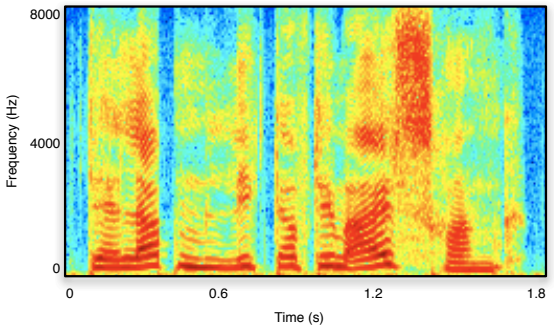


Figura 2.4: Ejemplo de señal de audio utilizada del corpus EmoDB. Espectrograma de la señal remuestreada a 16 kHz.

Cuadro 2.2: Matriz de confusión para los tests conducidos para la percepción humana, generada a partir de las etiquetas disponibles para cada una de las grabacioens listadas en el conjunto Database of German Emotional Speech.

	F	D	H	N	S	A	B
Fear	.85	.04	.03	.03	.01	.04	.00
Disgust	.03	.79	.01	.04	.08	.02	.02
Happiness	.02	.02	.83	.06	.01	.05	.06
Neutral	.00	.00	.01	.87	.04	.02	.06
Sadness	.06	.02	.00	.06	.78	.00	.08
Anger	.01	.01	.01	.01	.00	.96	.00
Boredom	.00	.01	.00	.00	.11	.03	.85

Como ya se ha mencionado anteriormente, los conjuntos de datos descritos representan emociones actuadas y, por tanto, existen diferencias con respecto a situaciones con emociones reales. No obstante, estos datos son ampliamente utilizados por la comunidad investigadora, por lo que proporcionan una buena referencia para este estudio.

3 Métodos

Debido a las diferencias en la naturaleza de los problemas planteados en este estudio, una serie de rasgos, algoritmos y técnicas de clasificación son utilizadas de forma integrada. Este capítulo proporciona una descripción teórica de todos los métodos y procedimientos utilizados durante los experimentos con más grado de detalle para aquellos que representan una innovación en el campo de estudio.

3.1. Rasgos

Una elección específica de rasgos para cada propósito parece adecuada para mejorar los resultados, si los comparamos con aquellos obtenidos con grupos de rasgos estándar diseñados para distintos propósitos. En trabajos similares a este, se concluye que distintas combinaciones de rasgos de audio funcionan bien en problemas de clasificación de datos de audio (Li et al., 2001). No obstante, durante este estudio se ha realizado un análisis específico de las características de la voz y su cuantificación para lograr una mejor comprensión de la naturaleza de las emociones humanas y de los artefactos utilizados para reconocerlas (Scherer et al., 2003; Banse and Scherer, 1996). Dadas las características de los conjuntos de datos utilizados, los grupos de rasgos elegidos para este estudio son descritos en esta sección.

3.1.1. Mel-frequency cepstral coefficients, (MFCC, ΔMFCC)

Los MFCC son extensamente utilizados en la comunidad investigadora del campo del reconocimiento automático de voz (ASR por sus siglas en inglés) en aplicaciones de distinta naturaleza (Fang et al., 2001; Logan, 2000). Sus derivadas de primer orden (ΔMFCC) también se usan con frecuencia puesto que son más robustas frente a efectos de ruido. Los MFCCs se obtienen a partir de una representación del espectro de potencia en la escala Mel. Ésta es una escala de frecuencias donde las distancias entre frecuencias percibidas por los humanos siguen una distribución logarítmica con respecto a la escala en Herzios. Es

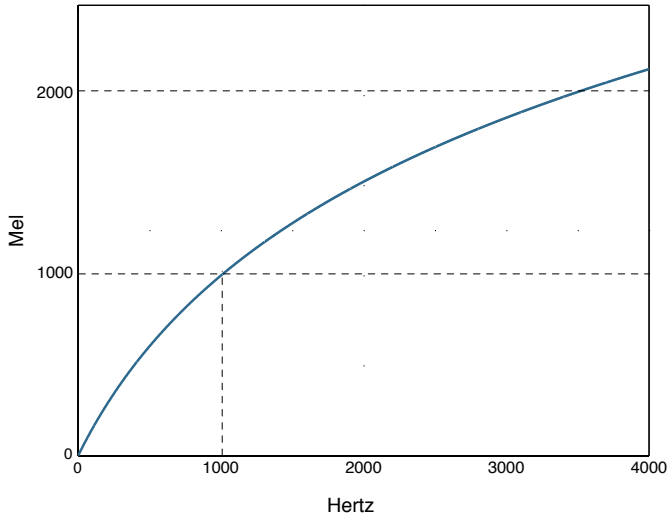


Figura 3.1: Representación de la escala Mel con respecto a la escala Hertz.

normalmente aceptado que la escala Mel es representada por la expresión dada en la Ecuación 3.1.

$$m = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \tag{3.1}$$

A modo de referencia, se establece la correspondencia de 1000 Herzios con 1000 Mels. Una representación de la relación Mel-Herzio se puede observar en la figura 3.1.

El Mel-frequency cepstral cepstrum representa la energía de la señal en diferentes bandas de frecuencia, uniformemente distribuídas en la escala Mel. Los coefficientes (MFCCs) son obtenidos siguiento estos pasos:

1. Calcular la DFT de la señal enventanada y obtener su espectro de potencia.
2. Aplicar un banco de filtros triangulares equiespaciados en la escala Mel.
3. Obtener la energía del log-espectro en cada banda.
4. Calcular la transformada discreta del coseno (DCT).

La DCT utilizada en estos pasos es una transformada basada en la suma de funciones coseno, utilizada normalmente para compresión de datos. Existen dis-

tintas variantes e implementaciones de ella pero, para este estudio, la expresión presentada en la Ec. 3.2 es utilizada:

$$MFCC_i = \sum_{k=1}^{K} X_k \cos\left(i\left(k-\frac{1}{2}\right)\frac{\pi}{K}\right), i = 1, ..., M \tag{3.2}$$

donde M, K y X_k representan el número de coeficientes MFCC, el número de bandas consideradas y la energía del log-espectro en la banda k-ésima respectivamente. Una representación del proceso para extraer los MFCCs se muestra en la Figura 3.2.

3.1.2. Rasgos de la Modulación Espectral

Basados en la arquitectura de extracción de los MFCC, estos parámetros representan el espectro demodulado a diferentes frecuencias. Lo que se pretende obtener es una medida de a qué velocidad y en qué cantidad varía el espectro con el tiempo. Diferentes bandas del espectro pueden contener información sobre distintas propiedades de la voz y, por tanto, el análisis por separado de éstas probablemente producirá un conjunto de rasgos que aporte gran cantidad de información (Scherer et al., 2003). Una secuencia temporal de la energía en cada banda es calculada y un análisis de Fourier de segundo nivel es realizado con ésta. De este modo, se llega a coeficientes para una banda concreta que miden la cantidad de cambios en la energía de dicha banda con respecto a las variaciones totales. La figura 3.3 explica en detalle cómo se obtienen los rasgos descritos, siguiendo estos pasos:

1. Transformada de Fourier de Tiempo Reducido (STFT por sus siglas en inglés) de la señal de audio.
2. Filtrado paso banda de cada enventanado con filtros equiespaciados en la frecuencia Mel.
3. Energía del log-espectro en cada banda, para cada enventanado.
4. Reconstruir la secuencia de las energías de cada banda.
5. Cálculo de la transformada de Fourier para cada secuencia de energías.
6. Cálculo de la energía del logaritmo de cada una de las señales obtenidas mediante la transformada de Fourier.
7. Obtención de la relación de energía en cada banda sobre la total, produciendo un coeficiente por cada una de las bandas definidas.

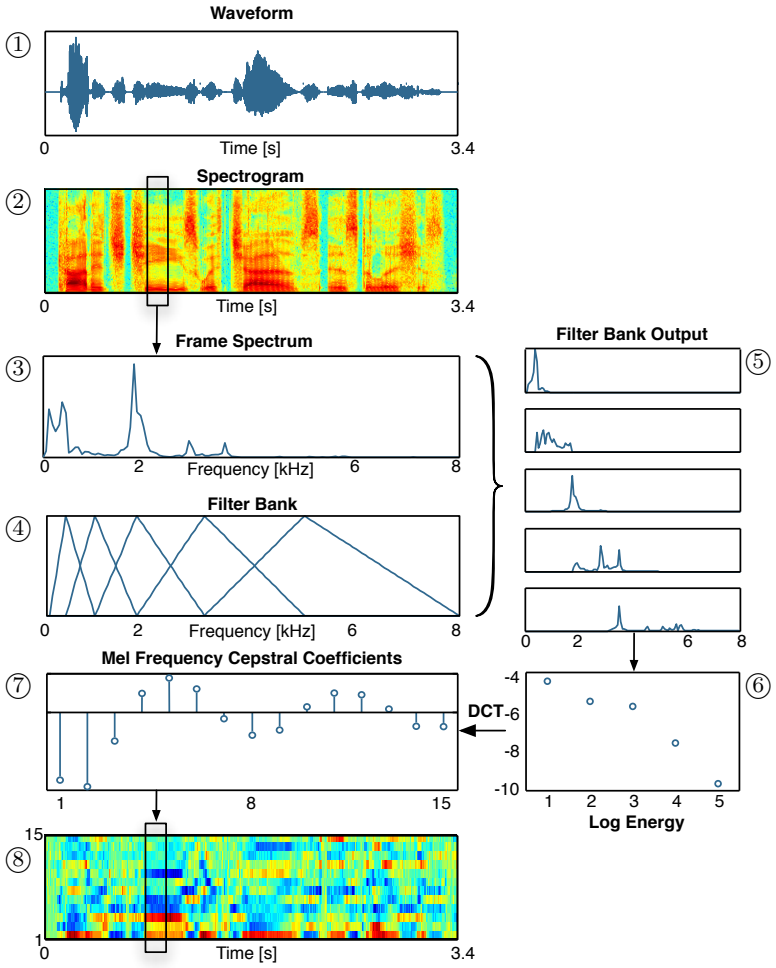


Figura 3.2: Algoritmo de extracción de MFCCs. A partir de la señal de voz ①, la transformada de Fourier en tiempo reducido (STFT) es calculada ②. El espectro de cada ventana ③ es pasado por un banco de filtros triangulares ④ igualmente espaciados en la escala de frecuencias Mel (ver Figura 3.1). Para cada señal filtrada paso-banda ⑤, la energía del logaritmo del espectro es calculada ⑥ y la transformada discreta del coseno (DCT) de estos valores representa los coeficientes MFCC de la ventana considerada ⑦. La concatenación de MFCCs sobre todas las ventanas conforma el conjunto de rasgos MFCC de la muestra de voz completa ⑧.

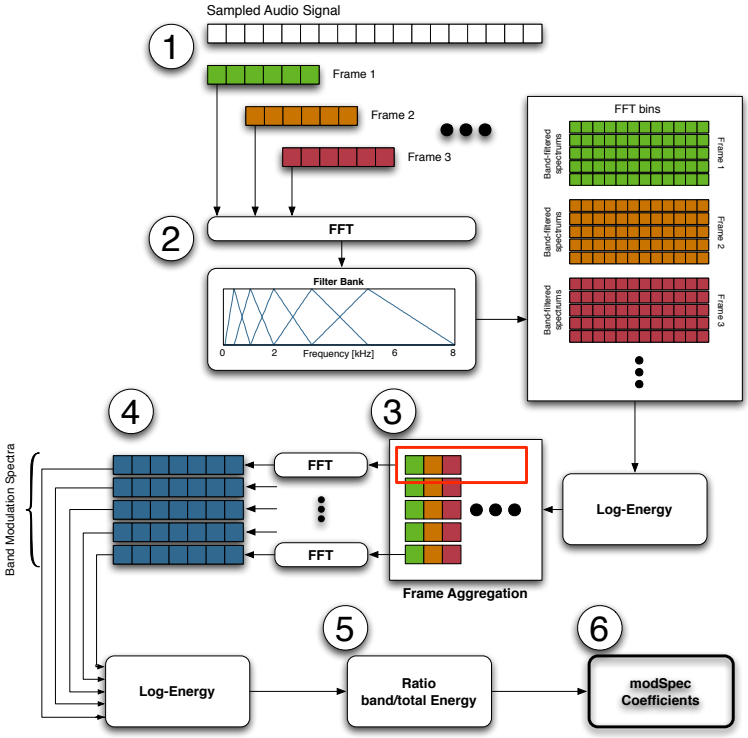


Figura 3.3: Algoritmo de extracción de los rasgos Modulación espectral. A partir de la señal de voz muestreada ① se obtiene la transformada rápida de Fourier (FFT) para cada ventana. El espectro de cada ventana se pasa por un banco de filtros triangulares ② igualmente espaciados sobre la escala de frecuencias Mel (ver Figura 3.1). Para cada ventana las energías paso-banda del log-espectro son calculadas. Una vez obtenidas, se concatenan ventana tras ventana, de modo que se consiguen secuencias de energía para cada banda ③. Para cada banda de frecuencia, una nueva FFT es calculada ④. Una vez más, la energía del logartimo del espectro obtenido es calculada para cada banda, junto con el ratio de cada una de ellas con respecto a la total ⑤. Los ratios se consideran directamente los coeficientes de los rasgos Modulación espectral ⑥.

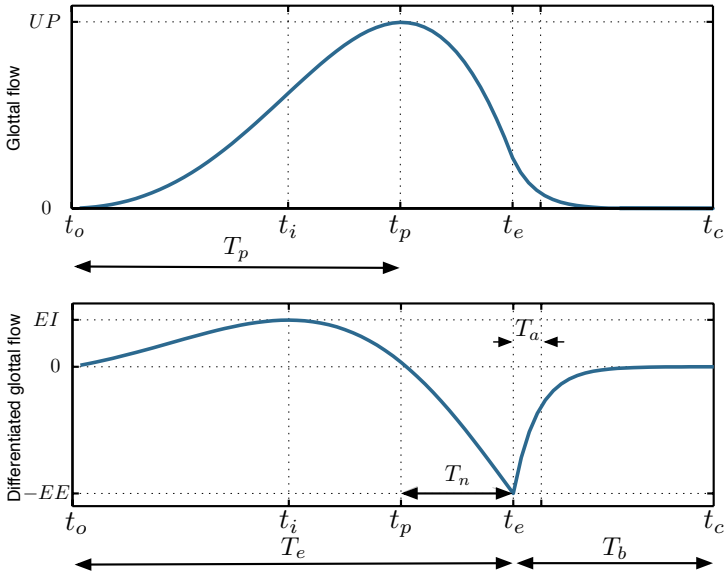


Figura 3.4: Ejemplo de flujo glotal (arriba) y su derivada (abajo) en el modelo de Liljencrants-Fant (LF).

3.1.3. Frecuencia Fundamental, f_0

Es posible obtener diferentes valores de frecuencia fundamental (f_0) en cada ventana. f_0 hace referencia a la componente frecuencial que porta la mayor parte de la energía dentro de la ventana considerada. A partir de la secuencia de valores de f_0, distintas medidas estadísticas son calculadas: media, desviación típica, máximo y cuartiles. Estos valores son los que unidos componen este conjunto de rasgos. La obtención de f_0 ha sido realizada mediante f_0 tracker, disponible en el paquete de software ESPS/waves+.

3.1.4. Calidad de Voz

El conjunto de rasgos que representa la calidad de voz está compuesto por un grupo de parámetros para los cuales es comúnmente aceptado que son responsables de los diferentes estilos de voz (Kane et al., under review, 2010; Gobl et al., 2002; Yanushevskaya et al., 2008). Estos rasgos pueden ser estimados a partir de la señal de excitación de la glotis, normalmente modelada con el modelo Liljencrants-Fant (LF), tal y como se muestra en la Figura 3.4.

Una completa descripción del modelo LF queda fuera de nuestro alcance en este proyecto, pero una definición de los parámetros utilizados para conformar el

conjunto de rasgos es dada. Una descripción más completa se puede encontrar en Scherer et al. (under review).

- EE: máxima velocidad de cierre de la glotis.
- T_p: tiempo transcurrido desde el momento en que la glotis comienza a cerrarse hasta que la máxima apertura es alcanzada.
- f_0: frecuencia fundamental, tal y como se describió en 3.1.3.
- T_e: tiempo transcurrido desde el momento en que la glotis comienza a abrirse hasta que se alcanza EE.
- T_a: fracción $\frac{1}{e}$ del tiempo transcurrido desde que se alcanza EE hasta que la glotis recupera su estado de relajación.

Una vez definidos estos parámetros del modelo, los valores que usados como rasgos pueden ser obtenidos utilizando las Ecuaciones 3.3, 3.4 y 3.5:

$$Rg = \frac{1}{2Tp \cdot f_0} \tag{3.3}$$

$$Rk = \frac{T_e - T_p}{T_p} \tag{3.4}$$

$$Ra = T_a \cdot f_0 \tag{3.5}$$

El conjunto de rasgos estará compuesto por la combinación de EE, R_g, R_k y R_a.

3.1.5. Energía

La energía es calculada usando ventanas de 32 ms, con un solapamiento de 16 ms. Para cada ventana n, la siguiente fórmula es utilizada para calcular la energía:

$$E_n = \frac{1}{W} \sum_{w=1}^{W} x^2[w] \tag{3.6}$$

donde $x[w]$ y W representan la señal y el tamaño de la ventana respectivamente. Estadísticos similares a los obtenidos para f_0 son utilizados para crear este conjunto de rasgos.

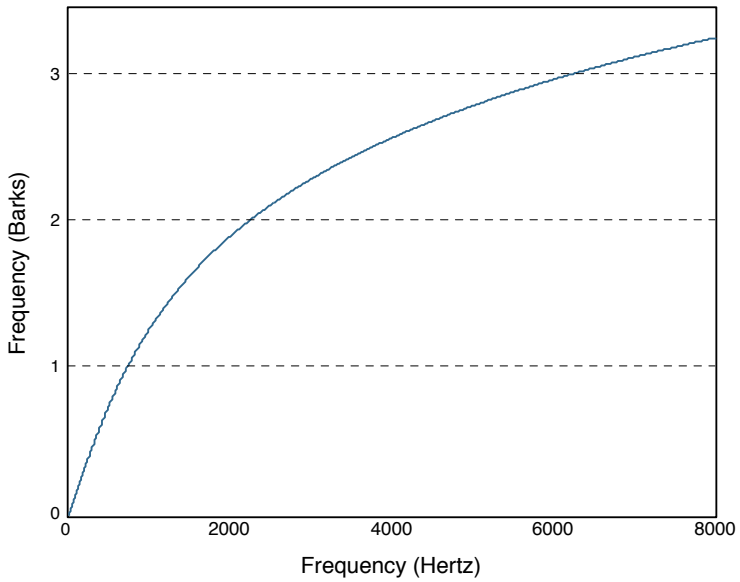


Figura 3.5: Relación Barks - Hertz, tal y como viene dada por la Ec. 3.7.

3.1.6. Análisis Perceptivo Mediante Predicción Lineal, PLP

Este conjunto de rasgos está basado en el modelo autoregresivo all-pole (sólo polos), tradicionalmente utilizado por la comunidad científica en el campo de ASR para estimación del espectro de potencia en tiempo reducido. Este modelo, obtenido mediante un análisis de predicción lineal es capaz de estimar el espectro en aquellas frecuencias con una alta energía, normalmente correspondientes con las formantes del tracto vocal. Sin embargo, el espectro es aproximado con una misma precisión para todas las bandas de frecuencia, lo cuál no se corresponde con las capacidades auditivas humanas. Los rasgos PLP fueron diseñados con la intención de representar mejor los niveles de amplitud percibidos, así como las distintas resoluciones espectrales sobre todas las bandas de frecuencias, tal y como fue descrito en Hermansky (1990), Hermansky and Morgan (1994). De un modo similar a la extracción de MFCCs, un banco de filtros es utilizado para un análisis por bandas. Estos filtros, sin embargo, no están igualmente espaciados en la frecuencia Mel, sino en la Bark, la cuál viene dada por la Ecuación 3.7. Su representación puede ser observada en la Figura 3.5.

$$b = 6 \cdot \mathrm{arcsinh}\left(\frac{f}{600}\right) \tag{3.7}$$

Una representación del proceso de extracción de los rasgos PLP puede observarse en la Figura 3.6.

Los pasos requeridos para la extracción de PLP son los siguientes (tal y como se explican en Hermansky (1990)).

1. STFT de la señal de audio.
2. Análisis de bandas críticas en la frecuencia Bark.
3. Pre-emphasis para igualar amplitudes.
4. Conversión intensidad-amplitud mediante compresión de raíz cúbica.
5. Transformada inversa de Fourier.
6. Solución de los coeficientes autoregresivos.

3.1.7. Periodicidad

Estudios previos sobre los datos utilizados mostraron que algunas de las expresiones pueden ser clasificadas si se considera la longitud de los segmentos de audio como un rasgo (Wendt, 2007). Puesto que nuestra intención es encontrar rasgos que puedan ser extraídos en cualquier contexto, el uso de esta propiedad resultaría en una mejora irreal de nuestros resultados. Por tanto, se ha desarrollado un nuevo conjunto de rasgos que pueda representar esta información, al menos parcialmente.

Considerar el número de sílabas por segundo no supone utilizar medidas inválidas puesto que puede ser estimado a partir de la señal directamente, en cualquier aplicación. Los detectores de sílabas pueden ser implementados de diferentes modos, tal y como se describe en Pfitzinger et al. (1996); Crystal and House (1990); Cedergren and Perreault (1994). La solución utilizada en este estudio es la descrita a continuación:

Asumamos que cada sílaba contiene al menos una vocal. Si se considera la alta periodicidad que caracteriza a las vocales en contraste con las consonantes, detecar segmentos de voz con una alta periodicidad nos proporcionaría un indicador de dónde hay una sílaba presente. Una solución inmediata para obtener un valor de periodicidad es calcular la función de autocovarianza en segmentos pequeños de la señal original. Una vez que este paso se ha completado, los segmentos pueden ser agrupados según su valor de periodicidad en una forma binaria: periódicos o no periódicos. Para lograr esto, un sistema con doble umbral ha sido diseñado para simular un ciclo de histéresis (Mayergoyz, 2003), tal y como puede observarse en la Figura 3.7. Este sistema marca el comienzo de una zona periódica cuando un valor por encima del 80 % del máximo es encontrado. De forma similar, el comienzo de una zona no periódica será detectado mediante la presencia de un valor por debajo del 30 % del máximo.

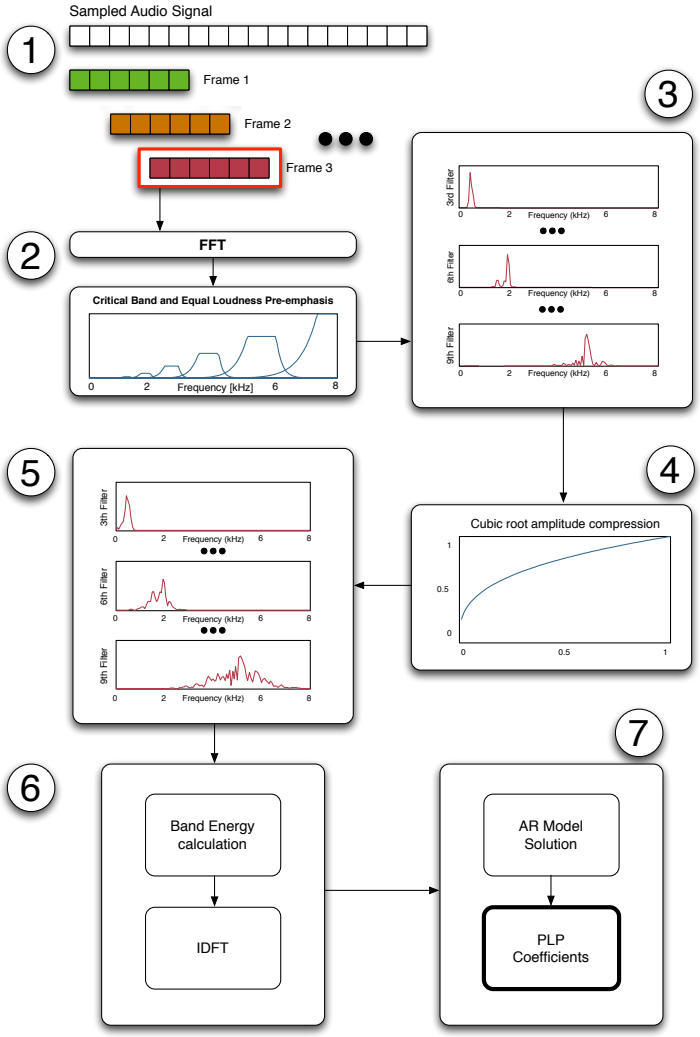


Figura 3.6: Algoritmo de extracción de los rasgos PLP. A partir de la señal de voz muestreada ① la transformada rápida de Fourier (FFT) es calculada en cada ventana. El espectro en cada ventana es filtrado con un banco de filtros ② igualmente espaciados en la escala de frecuencias Bark (ver Figura 3.5). Cada señal paso-banda ③ es comprimida en amplitud siguiendo una ley de raíz cúbica ④. A partir de los espectro comprimidos en amplitud ⑤, la energía de su logaritmo es calculada en cada banda y su transformada inversa de Fourier (IDFT) es calculada ⑥. La solución del modelo autoregresivo (AR) es llevada a cabo, siendo los valores obtenidos los coeficientes del conjunto de rasgos PLP ⑦.

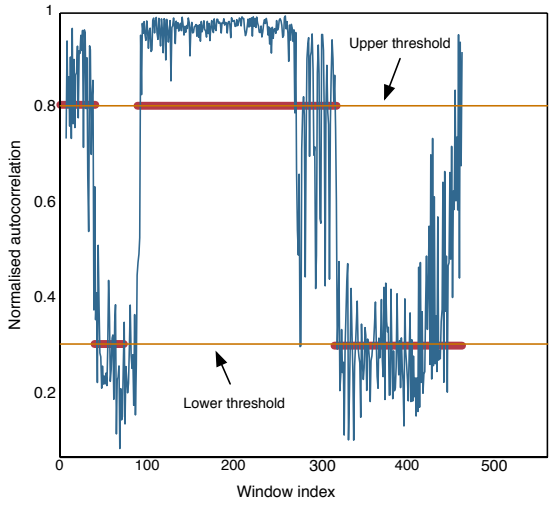


Figura 3.7: Conjunto de rasgos de Periodicidad. En azul: función de autocorrelación sobre ventanas temporales consecutivas (cada 5ms). En naranja: umbrales superior e inferior para identificar los estados binarios. En rojo: estados detectados, valores altos y bajos representan segmentos periódicos y no periódicos respectivamente.

Adicionalmente, las variaciones de energía también son utilizadas para detectar sílabas. Asumiendo menores energías en los bordes entre sílabas, es posible utilizar un detector de envolvente y, una vez más, un sistema con doble umbral para detectar las sílabas. Este proceso se muestra en la Figura 3.8.

Con el fragmento de voz inicial dividido en segmentos periódicos y no periódicos así como en zonas de alta o baja energía, se calcula: las longitudes de cada parte, la mayor diferencia en longitud y la relación de energía entre cada una de las partes con respecto a la longitud total. La combinación de estos valores conforma el conjunto de rasgos completo.

3.2. Datos Secuenciales

El análisis de series temporales es una tarea compleja debido a las propiedades secuenciales de los datos. Debido a la dependencia temporal, las observaciones no se pueden considerar estadísticamente indpendientes y, por tanto, un modelo complejo capaz de tratar con este tipo de datos es necesario.

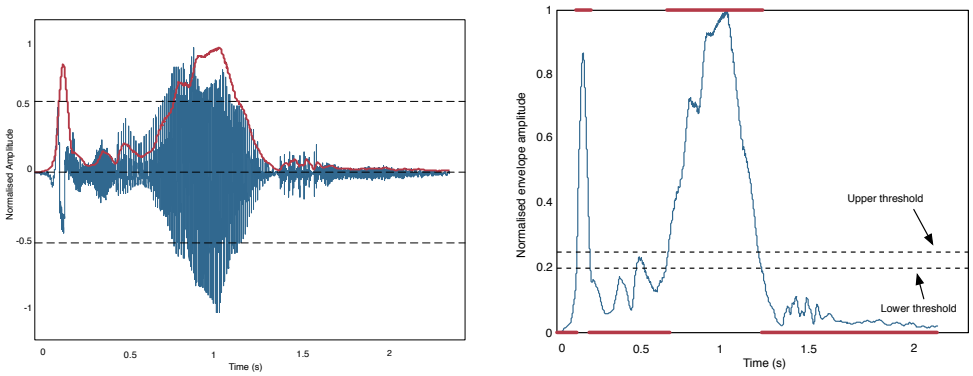


Figura 3.8: A la izquierda: datos normalizados (en azul) y envolvente detectada (en rojo). A la derecha: envolvente detectado (en azul), valores de umbral para identificar los estados binarios (línea discontinua), y segmentos para los distintos estados detectados (en rojo).

3.2.1. Modelos de Markov ocultos

Un modelo estadístico comúnmente utilizado para tratar datos secuenciales son los modelos de Markov ocultos (HMM por sus siglas en inglés). Éstos son supuestos modelos de Markov con estados no observables. Un modelo de Markov es un modelo estocástico basado en la suposición de no tener memoria. De acuerdo con esta propiedad, los estados futuros dependen únicamente del estado actual y no en la historia del sistema, tal y como se describe en las Ecuaciones 3.8 y 3.9.

$$p(x_n \mid x_1, ..., x_{n-1}) = p(x_n \mid x_{n-1}) \tag{3.8}$$

$$p(x_1, ..., x_N) = \prod_{n=1}^{N} p(x_n \mid x_1, ..., x_{n-1}) = p(x_1) \prod_{n=2}^{N} p(x_n \mid x_{n-1}) \tag{3.9}$$

donde x_n representa el estado del model en el instante n. También existen extensiones de mayor orden para los modelos de Markov, donde no sólo el estado actual es considerado para predecir los estados futuros, sino que también los $m-1$ estados previos son considerados, donde m representa el órden del modelo. Esta extensión, sin embargo, no es relevante para nuestro trabajo y será por tanto omitida. En los modelos de Markov los estados son visibles y los únicos parámetros del sistema son las probabilidades de transición. En el caso de los HMM, los estados son observables sólo parcialmente. Las observaciones están vinculadas a los estados del sistema pero no son suficientes para especificarlos. Los HMM pueden ser definidos en función de los siguientes elementos:

1. Número de estados, N: representa el número de estados ocultos que se asumen para el HMM.

2. Distribución de probabilidades para la transición entre estados, A: existen una serie de probabilidades asociadas con cada uno de los estados que define la transición entre éstos en cada instante.

3. Función densidad de probabilidad de las observaciones (fdp), b: representa la fdp de las emisiones de cada estado. Dado un estado i, ésta es la distribución que genera las observaciones. Puede ser considerada una distribución discreta o contínua, dependiendo del tipo de observaciones que se quieran modelar.

4. Distribucion de probabilidad de estados iniciales π: para el caso de no existir estados previos.

En la literatura existen tres problemas típicos de uso de HMMs, dependiendo del tipo de tarea que se requiere realizar con ellos.

- Problema 1: Dada una secuencia de observaciones O y el HMM λ, calcular la verosimilitud (likelihood en inglés) $P(O \mid \lambda)$.

- Problema 2: Dada una secuencia de observaciones O y el HMM λ, obtener la secuencia de estados Q que mejor explica las observaciones.

- Problema 3: Dada una secuencia de observaciones O, ajustar los parámetros del modelo λ para maximizar la verosimilitud $P(O \mid \lambda)$.

Debido al interés de este trabajo, solamente nos centraremos en los problemas 1 y 3, normalmente conocidos como evaluación y entrenamiento, respectivamente.

3.2.1.1. Problema de evaluación

La verosimilitud $P(O \mid \lambda)$ se define como la probabilidad de la secuencia de observaciones O, dados los parámetros del modelo, λ. Proporciona un modo de medir cómo de bien puede explicar el modelo las observaciones. Su logaritmo (log-likelihood) se utiliza frecuentemente ya que representa una función igualmente monótona y su obtención es más sencilla, al convertir multiplicaciones en sumas. Puesto que la secuencia real de estados no se puede observar, existen muchas secuencias diferentes que podrían haber generado unas observaciones dadas. El cálculo mediante fuerza bruta de todas las posibles secuencias de estados de longitud T implicaría un número de operaciones extramadamente grande (del orden de $2T \cdot N^T$). Por tanto, un método más práctico ha de ser considerado. El procedimiento utilizado para este propósito es el llamado procedimiento forward (avance).

Asumamos una secuencia de observaciones $O = \{O_1, O_2, ..., O_T\}$ de longitud T y un HMM de parámetros λ. Es posible definir la variable de avance $\alpha_t(i) = P(O_1, O_2, ..., O_t, q_t = S_i \mid \lambda)$ como la probabilidad conjunta de las observaciones hasta el instante t y estar en el estado S_i en el mismo instante t, dado λ. $P(O \mid \lambda)$ puede ser obtenida entonces siguiendo recursivamente los siguientes pasos:

- Paso 1: $\alpha_1(i) = \pi_i b_i(O_1) \quad \forall i \in 1, ..., N.$
- Paso 2: $\alpha_{t+1}(j) = \left(\sum_{i=1}^{N} \alpha_t(i) a_{ij} \right) b_j(O_{t+1}), \quad \forall t \in 1, ..., T-1.$
- Paso 3: $P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i).$

donde a_{ij} corresponde con el elemento (i, j) de A y representa la probabilidad de transición del estado S_i al estado S_j en cualquier instante de tiempo. $b_j(O_t)$ representa el valor de la j-ésima fdp, dado el símbolo observado O_t.

3.2.1.2. Problem de entrenamiento

No existe un modo analítico para caclular los parámetros óptimos de un modelo HMM. Sin embargo, sí existen algoritmos iterativos que pueden estiamarlos de manera localmente óptima. El método más utilizado es del denominado Baum-Welch, equivalente al Expectation-Maximzation (EM) en este caso.

En primer lugar, variables de avance y retroceso (forward-backward) deben considerarse:

$\alpha_t(i)$: como se definió en la Sección 3.2.1.1.

$\beta_t(i)$: de forma similar a $\alpha_t(i)$, es posible definir una variable de retroceso que represente la probabilidad conjunta de las observaciones a partir del instante t hasta el instante final T, dados los parámetros λ y el estado en el instante t, S_i:

$$\beta_t(i) = P(O_{t+1}, ..., O_T \mid q_t = S_i, \lambda) \tag{3.10}$$

Una vez más se puede realizar inducción para obtener los valores de β en cada instante:

- Step 1: $\beta_T(i) = 1, \quad \forall i \in 1, ..., N$
- Step 2: $\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j), \quad \forall t \in 1, ..., T-1; \quad \forall i \in 1, ..., N$

A partir de las definiciones de $\alpha_t(i)$ y $\beta_t(i)$, nuevas variables pueden ser definidas para llevar a cabo el proceso de Baum-Welch:

$\xi_t(i,j)$: representa la probabilidad de estar en el estado S_i en el instante t y en el estado S_j en el instante $t+1$, dada la secuencia de observaciones O y los parámetros del modelo λ:

$$\xi_t(i,j) = P(q_t = S_i, q_{t+1} = S_j \mid O, \lambda) \tag{3.11}$$

$\gamma_t(i)$: probabilidad de estar en el estado S_i en el instante t, dada la secuencia de observaciones O y los parámetros del modelo λ:

$$\gamma_t(i) = P(q_t = S_i \mid O, \lambda) \tag{3.12}$$

Utilizando las definiciones anteriores, ahora debemos expresar estas variables en términos de los parámetros conocidos:

$$\begin{aligned}
\xi_t(i,j) &= \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{P(O \mid \lambda)} \\
&= \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N} \alpha_t(i) \beta_t(i)} \\
&= \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N} \alpha_t(i) \left[\sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j) \right]} \\
&= \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}
\end{aligned} \tag{3.13}$$

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j) \tag{3.14}$$

La combinación de las expresiones obtenidas puede ser utilizada para estimar ciertas probabilidades del modelo. Definamos τ_i como el número de transiciones iniciadas en el estado S_i y τ_{ij} como el número de transiciones desde el estado S_i hacia el estado S_j. Entonces es posible definir:

Esperanza del número de transiciones iniciadas en el estado S_i:

$$E\left[\tau_i\right] = \sum_{t=1}^{T-1} \gamma_t(i) \tag{3.15}$$

Y esperanza del número de transiciones desde el estado S_i al estado S_j:

$$E\left[\tau_{ij}\right] = \sum_{t=1}^{T-1} \xi_t(i,j) \tag{3.16}$$

Finalmente, expresiones para los nuevos parámetros del modelo pueden ser obtenidas:

$$\bar{\pi}_i = \gamma_1(i) \tag{3.17}$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \tag{3.18}$$

$$\bar{b}_i(k) = \frac{\sum_{t=1 \cap O_t = v_k}^{T} \gamma_t(i)}{\sum_{t=1}^{T} \gamma_t(i)} \tag{3.19}$$

donde v_k representa el símbolo emitido en el instante t por la distribución discreta $b_j(k)$. Las expresiones para re-estimar los parámetros (Ecs. 3.17, 3.18 y 3.19) se corresponden exactamente con la solución obtenida mediante EM en este problema particular.

Hasta este punto las distribuciones de emisión de observaciones han sido consideradas discretas. Esto, sin embargo, no es una suposición que se cumpla en muchas aplicaciones, donde las observaciones no son discretas, sino continuas. Para estas situaciones modificaciones han de ser incluidas en la formulación para modelar las observaciones como mezclas de distribuciones Gausianas continuas.

Las probabilidades de transición entre estados no ven afectada su definición debido a esta extensión, pero las distribuciones de emisión de símbolos deben ser consideradas como:

$$b_j(O) = \sum_{m=1}^{M} c_{jm} \Psi(O, \mu_{jm}, U_{jm}), 1 \leq j \leq N \tag{3.20}$$

donde O representa la variable que queremos modelar, c_{jm} es el coeficiente de mezclado de la m-ésima componente del estado j. Ψ hace referencia a cualquier densidad, pero será considerada como una distribución Gausiana para el propósito de este estudio, puesto que es una distribución bien conocida y puede ser usada para aproximar cualquier función de densidad continua. μ_{jm} y U_{jm} representan la media y la matriz de covarianza de las correspondientes componentes.

Los coeficientes de mezclado están sujetos a las siguientes restriccciones:

$$\sum_{m=1}^{M} c_{jm} = 1, 1 \leq j \leq N \tag{3.21}$$

$$c_{jm} \geq 0, 1 \leq j \leq N, 1 \leq m \leq M \tag{3.22}$$

No es complicado obtener las formulas de re-estimado para los parámetros de la distribución. La esperanza del número de ocurrencias del estado j con la componente k activa, con respecto del total de ocurrencias del estado j es utilizada para estimar los coeficientes de mezclado:

$$\bar{c}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k)}{\sum_{t=1}^{T} \sum_{k=1}^{M} \gamma_t(j,k)} \tag{3.23}$$

De forma similar a la Ec. 3.23, pero ponderando cada numerador por el valor de las observaciones, es posible obtener un estimador del valor producido por la k-ésima componente, el cual puede ser utilizado como re-estimación de la media μ_{jk}:

$$\bar{\mu}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) O_t}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{3.24}$$

Una vez más, para la reestimación de la matriz de covarianza U_{jk}, se puede encontrar una expresión similar:

$$\bar{U}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k)(O_t - \mu_{jk})(O_t - \mu_{jk})\prime}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{3.25}$$

donde

$$\gamma_t(j,k) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)} \right] \left[\frac{c_{jk}\Psi(O_t, \mu_{jk}, U_{jk})}{\sum_{m=1}^{M} c_{jm}\Psi(O_t, \mu_{jm}, U_{jm})} \right] \tag{3.26}$$

Las expresiones obtenidas permiten una actualización iterativa de los parámetros de los HMM, hasta que se alcance cierto nivel de convergencia.

3.2.2. Normalización de los datos

Puesto que el entrenamiento de los HMMs se realiza mediante una maximización local y dado que los espacios de rasgos son generalmente muy heterogéneos, el modelo obtenido es muy susceptible a diferentes inicializaciones. Por tanto, un uso directo de los rasgos para entrenar los HMMs puede resultar una tarea complicada si los parámetros iniciales del modelo no son elegidos apropiadamente. El efecto que puede ocurrir en este caso es el mostrado en la Figure 3.9. Las Gausianas son demasiado estrechas para el espacio de los rasgos y el algoritmo de entrenamiento, probablemente, no funcionará bien.

En nuestros experimentos, un proceso de normalización es llevado a cabo con antelación al entrenamiento de los HMMs, para evitar este efecto. La normalización es realizada mediante la Ecuación 3.27, donde x representa los datos a normalizar y μ, σ, su media y desviación típica respectivamente. Durante el entrenamiento del sistema, la media y desviación típica (μ_{train} y σ_{train}) son calculadas en el dominio de cada grupo de rasgos y para cada clase, antes de entrenar los HMMs. Para eliminar el efecto de valores atípicos (outliers), todos los valores por encima y por debajo de los percentiles al 95 % y 5 % respectivamente son descartados. Con los datos normalizados, los HMMs son entrenados y los mismos valores de normalización son utilizados más adelante para normalizar las muestras desconocidas en el proceso de test, antes de calcular sus valores de verosimilitud. El efecto de la normalización puede observarse en la Figura 3.10.

$$x_{norm} = \frac{x - \mu}{\sigma} \tag{3.27}$$

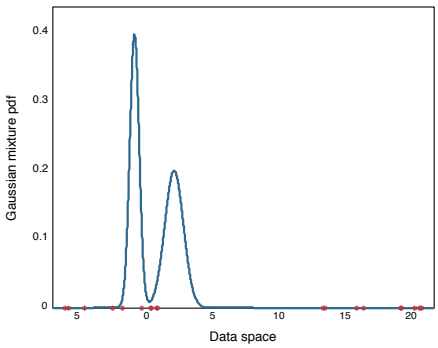


Figura 3.9: Ejemplo de datos y modelo de Gausianas mezcladas (GMM). Los datos son generados siguiento la distribución $x \sim U[12:21] + U[-7:1]$. El modelo mezclado está compuesto por dos Gausianas con parámetros $\mu_1 = -1, \sigma_1 = 0{,}5$ y $\mu_2 = 2, \sigma_2 = 1$ respectivamente. Es posible observar que una inicialización aleatoria de las Gausianas queda demasiado lejos de la distribución real de los datos, por lo que es improbable que el proceso de adaptación pueda llevarse a cabo debido a valores de verosimilitud demasiado bajos.

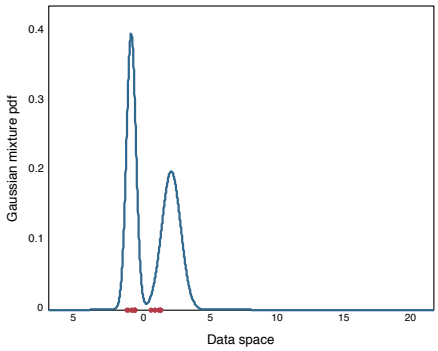


Figura 3.10: Datos normalizados y GMM inicial. Como puede verse, la normalización de los datos ha permitido que la mezcla de Gausianas produzca mejores valores de verosimilitud, lo que permitirá un mejor proceso de adaptación incluso desde las primeras iteraciones.

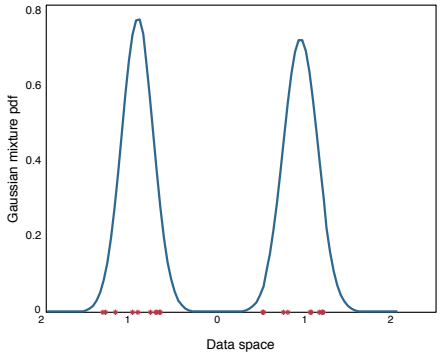


Figura 3.11: Datos normalizados y GMM adaptado. Tras el proceso de adaptación, el modelo GMM puede representar bien la distribución de los datos.

Una vez que los datos de entrenamiento han sido normalizados, es más probable que los parámetros iniciales de los HMMs sean correctamente adaptados y representen bien a distribución real que generó los datos. Un ejemplo de un modelo bien adaptado se puede observar en la Figura 3.11.

3.2.3. Alineamiento de los datos

Los modelos de Markov ocultos se utilizan frecuentemente para modelar datos secuenciales. No obstante, y puesto que este trabajo está enfocado a estudiar los datos y sus características para cada emoción por separado, un espacio de representación donde éstas puedan ser comparadas es necesario.

La clasificación de emociones a partir voz es un problema que presenta grandes retos debido a la naturaleza secuencial de los datos. Por tanto, rasgos dinámicos extraídos en segmentos cortos de audio (ventanas en torno a 32ms) son muy útiles para la clasificación de grabaciones con expresiones. Sin embargo, para ser capaces de comparar estos rasgos secuenciales con rasgos estáticos es necesario codificarlos en vectores de una longitud fija. Existen diferentes formas de abordar este tipo de situaciones. En este estudio, los HMMs son usados para codificar los datos secuenciales y transformarlos a un nuevo espacio de representación, donde cada secuencia puede ser representada en términos de un número fijo de dimensiones (Bicego et al., 2003). Los grupos de rasgos considerados secuanciales y, por tanto, alineados mediante este procedimiento son $MFCC$, $\Delta MFCC$, $ModulacionEspectral$ y $CalidaddeVoz$.

Asumamos un conjunto de observaciones de referencia $\mathcal{R} = \{O_1, ..., O_R\}$, donde O_i, $\forall i = 1, ..., R$ es una observación sin restricciones de longitud. Es posible entrenar, para cada una de las observaciones de referencia, un HMM the represente adecuadamente el modelo que la produjo. Por tanto, es sencillo obtener un conjunto $\lambda = \{\lambda_1, ..., \lambda_R\}$ de HMMs donde λ_i, $\forall i = 1, ..., R$ representa el

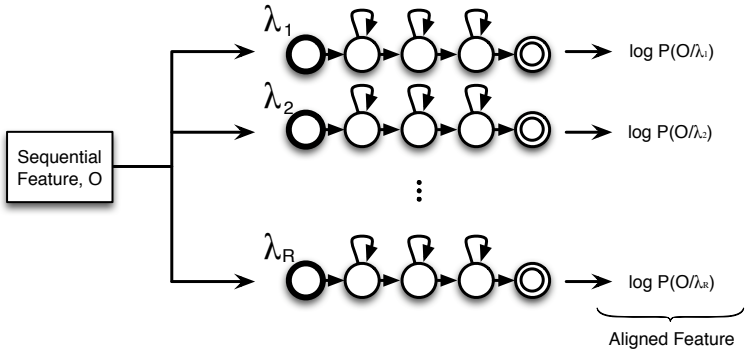


Figura 3.12: Esquema de alineamiento de rasgos. Una observación O de un rasgos secuencial es utilizada para calcular la verosimilitud de cada uno de los R HMMs entrenados. Los valores obtenidos de cada uno de ellos son unidos y considerados un único vector de dimensión R, conformando así un vector de longitud fija para cada observación.

HMM entrenado a partir de la secuencia O_i. Dada una observación secuencial $\tilde{O}$, su representación como un único vector en el espacio R-dimensional es obtenida calculado su log-likelihood con respecto a los HMM de referencia:

$$D_R(\tilde{O}) = \frac{1}{T} \times \begin{pmatrix} \log P(\tilde{O} \mid \lambda_1) \\ \log P(\tilde{O} \mid \lambda_2) \\ \vdots \\ \log P(\tilde{O} \mid \lambda_R) \end{pmatrix}, \tag{3.28}$$

donde T representa la longitud de la secuencia y $P(\tilde{O} \mid \lambda_i)$ es la verosimilitud del i-ésimo HMM, dada la observación $\tilde{O}$. Con esta transformación, podemos crear un nuevo espacio de dimensión R, donde cada secuencia observada es representada como un único vector. En este espacio euclídeo, cualquier técnica estándar de clasificación (aprendizaje supervisado, semisupervisado, clústers, etc) puede ser utilizado. Una representación del sistema de alineamiento se puede observar en la Figura 3.12.

Para los experimentos realizados en este estudio, un subconjunto fue aleatoriamente escogido entre los segmentos de audio iniciales para entrenar los HMMs. Para cada clase diferente c, $\forall c = 1, ..., C$, donde C representa el número de clases, 2 HMMs fueron entrenados. Puesto que los datos usados están compuestos por 6 clases distintas, un total de $R = 12$ HMMs fueron necesarios. Cada HMM fue inicializado con 2 estados y una mezcla de 2 Gausianas por estado. Cada modelo fue entrenado con 5 segmentos de audio diferentes durante un número de iteraciones no superior a 30. Experimentos con un mayor número de estados y componentes de mezclado no demostraron generar mejores resultados y, por el contrario, sí que requerían mucho más tiempo de cómputo.

3.3. Análisis de Componentes Principales

El uso de un conjunto grande de HMMs entrenados para el alineamiento de los datos conduce a un escenario con una alta dimensionalidad, donde la inspección de los datos no es una tarea trivial. Existen diferentes técnicas para reducir la dimensionalidad, permitiendo la transformación de los datos a un espacio con menores dimensiones. En este estudio, la técnica utilizada es el análisis de componentes principales (PCA por sus siglas en inglés), introducido inicialmente en Pearson (1901). PCA es un procedimiento mediante el cual se convierten variables correladas en otras variables incorreladas, las llamadas componentes principales (PC). Las componentes principales tienen la propiedad de que cada una de ellas es ortogonal con respecto a la componente anterior, siendo la primera componente una variable que tiene máxima varianza. El número total de componentes principales es menor o igual que las dimensiones de los datos originales. La implementación utilizada en estos experimentos está basada en la descomposición en autovalores, generando una matriz de transformación. Representación de la primera componente principal frente a la segunda componente principal es utilizada en algunas gráficas de este trabajo (eg. Figures 4.4, 4.5). Un ejemplo de la transformación que tiene lugar mediante PCA es mostrada en las Figuras 3.13 y 3.14. En estas figuras una distribución Gaussiana en 2 dimensiones con media $\mu = (3, 2)^T$ y matriz de covarianza $\sigma = \begin{pmatrix} 2 & -1{,}5 \\ -1{,}5 & 2 \end{pmatrix}$ es mostrada. Mediante PCA, la matriz de transfromación

$$\begin{pmatrix} 0{,}7087 & 0{,}7055 \\ -0{,}7055 & 0{,}7087 \end{pmatrix} \tag{3.29}$$

es obtenida. Las direcciones de las componentes principales también pueden observarse en el gráfico. Como se puede ver, los vectores obtenidos para la transformación representan las cirecciones de mayor varianza y son, al mismo tiempo, ortogonales entre sí.

3.4. Máquinas de vectores de soporte, SVM

Las máquinas de vectores de soporte (SVM por sus siglas en inglés), son uno de los métodos más comúnmente utilizados para todo tipo de problemas de clasificación. Asumiendo dos clases linealmente separables, lo que se pretende es encontrar un hiperplano que puede separarlas maximizando el margen entre los nodos que lo soportan, los vectores de soporte, tal y como se muestra en la Figura 3.15. Si la suposición de que las clases son linealmente separables no se cumple, es posible definir una transformación a un espacio de mayores dimensiones mediante el uso de una función kernel, convirtiendo la búsqueda del hiperplano en una tarea más sencilla. Existen versiones extendidas de los

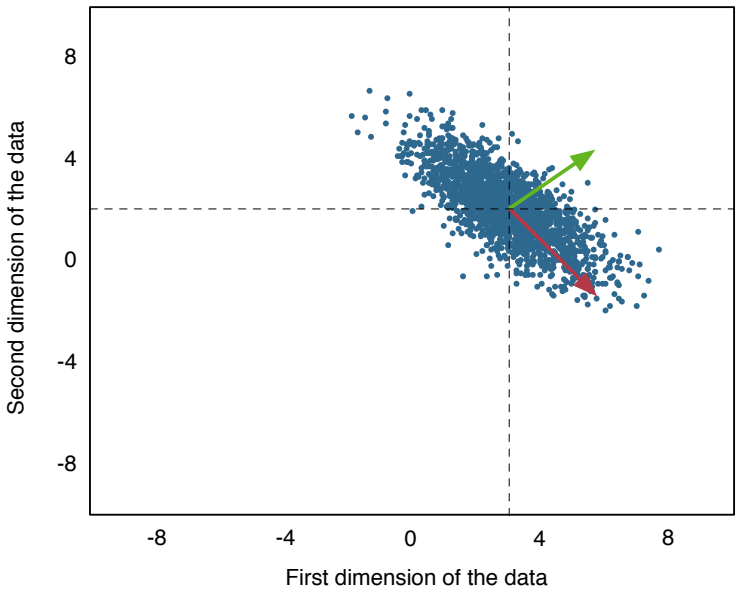


Figura 3.13: Ejemplo de análisis de componentes principales. En azul: distribución Gausiana bidimensional. En rojo: la primera componente principal representa la dirección de la variable con una mayor varianza. En verde: la segunda componente principal representa la dirección ortogonal a la primera, que presenta la mayor varianza posible.

SVMs para las situaciones en que más de dos clases han de ser separadas. Para el propósito de este estudio, SVMs en configuración uno-contra-uno para un problema de multi clases serán considerados (Kahsay et al., 2005).

Las implementaciones tradicionales de los SVMs están basadas en un escenario entrada rígida, salida rígida. Sin embargo, hay situaciones en las que una entrada con etiqueta rígida puede ser difícil de obtener debido a la percepción subjetiva de un anotador. Para este tipo de situaciones, SVMs difusos (FSVM) fueron diseñados, produciendo una salida rígida a partir de una entrada difusa. Aún más, un diseño posterior considerando entrada difusa y salida difusa (F^2SVM) también ha sido desarrollado (Thiel et al., 2007).

En esta sección tanto SVM clásicos como F^2SVM son explicados.

3.4.1. SVMs con entrada rígida y salida rígida

Un conjunto de entrenamiento inicial M con dos clases puede ser definido como:

$$M = \{(x_\mu, l_\mu) \mid x_\mu \in \mathbb{R}^n, l_\mu \in \{-1, +1\}, \quad \forall \mu = 1, ..., |M|\} \tag{3.30}$$

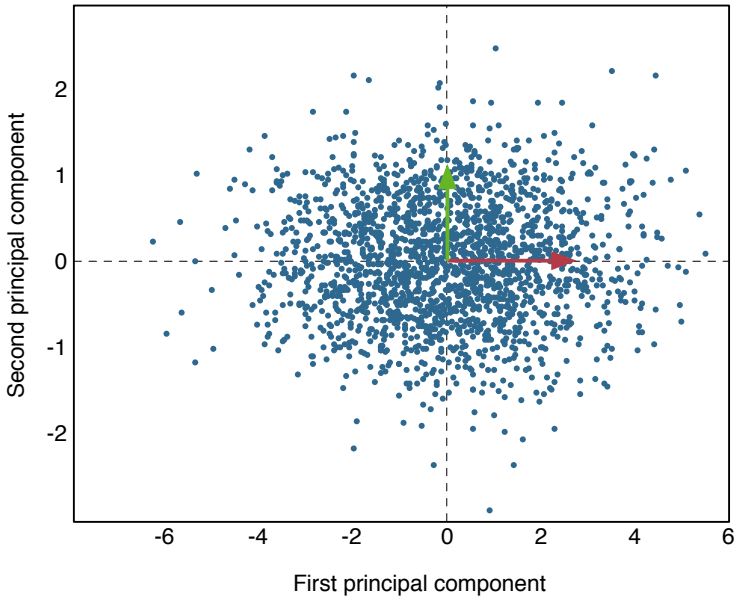


Figura 3.14: Ejemplo de análisis de componentes principales. En azul: distribución Gausiana bidimensional normalizada y transformada al espacio de las componente principales. Los ejes X e Y representan la primera y segunda componentes principales respectivamente, también representadas por las flechas roja y verde.

donde x_μ representa la μ-ésimo muestra y l_μ su correspondiente etiqueta. Es posible definir un hiperplano caracterizado por la normal a su superficie w y un valor de desplazamiento b que además satifaga la restricción:

$$l_\mu(w^T x_\mu + b) \geq 1, \quad \forall \mu = 1, ..., |M| \tag{3.31}$$

Tras la maximización del margen, al menos dos muestras deben cumplir la restricción de igualidad en la Ec. 3.31. Asumamos que estas muestras son x_ν y x_λ, pertenecientes a los subgrupos etiquetados como positivo y negativo respectivamente. Entonces, la anchura del margen puede ser expresada como:

$$\frac{w^T}{||w||}(x_\nu - x_\lambda) = \frac{2}{||w||} \tag{3.32}$$

Como se explica en Bishop (2006), maximizar la Ec. 3.32 es equivalente a minimizar:

$$\theta(w) = \frac{||w||^2}{2} \tag{3.33}$$

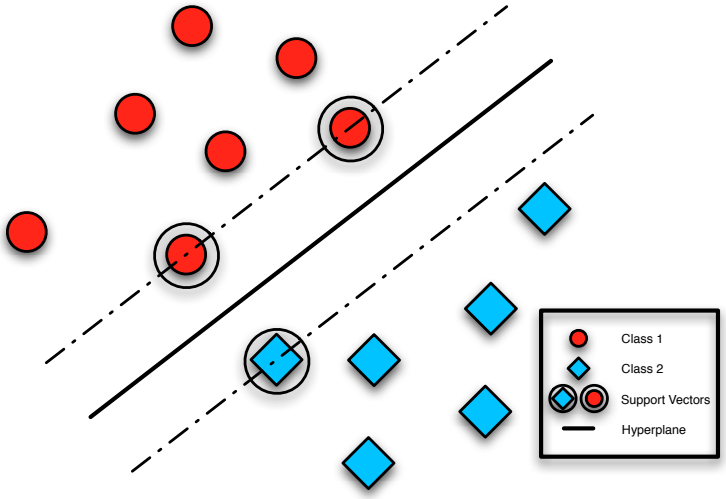


Figura 3.15: Ejemplo de máquina de vectores de soporte. Datos de dos clases están representadas (en rojo y azul respectivamente). El hiperplano de separación que maximiza la distancia entre ellos también se muestra. En este caso, los vectores de soporte son las tres muestras dentro de un círculo negro que coinciden con las líneas de punto y raya.

Para resolver este problema es necesario el uso de multiplicadores de Lagrange $\alpha_\mu \geq 0$, obteniendo la función:

$$L(w, b, \alpha) = \frac{||w||^2}{2} - \sum_{\mu=1}^{|M|} \alpha_\mu \{l_\mu (w^T x_\mu + b) - 1\} \tag{3.34}$$

Si las restricciones $\frac{\partial L}{\partial w} = 0$ y $\frac{\partial L}{\partial b} = 0$ son introducidas, las Ecs. 3.35 y 3.36 pueden ser deducidas.

$$w = \sum_{\mu=1}^{|M|} \alpha_\mu l_\mu x_\mu \tag{3.35}$$

$$0 = \sum_{\mu=1}^{|M|} \alpha_\mu l_\mu \tag{3.36}$$

Considerando este resultado, w y b pueden ser obviadas de la Ec. 3.34, conduciendo al problema de maximizar:

$$\tilde{L}(\alpha) = \sum_{\mu=1}^{|M|} \alpha_\mu - \frac{1}{2} \sum_{\nu=1}^{|M|} \sum_{\mu=1}^{|M|} \alpha_\nu \alpha_\mu l_\nu l_\mu x_\nu^T x_\mu \tag{3.37}$$

sujeto a las restricciones:

$$\alpha_\mu \geq 0, \quad \forall \mu = 1, ..., |M| \tag{3.38}$$

$$\sum_{\mu=1}^{|M|} \alpha_\mu l_\mu = 0 \tag{3.39}$$

Suponiendo que se alcanza una solución que maximiza el margen, también se cumplen las condiciones de Karush-Kuhn-Tucker (Bishop, 2006):

$$\alpha_\mu \{l_\mu (w^T x_\mu + b) - 1\} = 0, \quad \forall \mu = 1, ..., |M| \tag{3.40}$$

y si para todo $\alpha_\mu \neq 0$ se verifica:

$$l_\mu (w^T x_\mu + b) = 1 \tag{3.41}$$

siendo x_μ un vector de soporte. Entonces, para un nuevo punto x, la clasificación en una clase u otra puede ser calculada como:

$$y(x) = sign \left(\sum_{\mu=1}^{|M|} \alpha_\mu l_\mu x^T x_\mu + b \right) \tag{3.42}$$

El desplazamiento b puede ser determinado con cualquier vector de soporte x_ν mediante la expresión:

$$b = \frac{1}{l_\nu} - w^T x_\nu \tag{3.43}$$

Hasta este punto hemos asumido que existe un hiperplano de separación. Sin embargo, en una situación más realista, esta suposición no tiene por qué cumplirse y, por tanto, cierta reformulación es necesaria. Las variables de holgura ξ_μ pueden introducirse para permitir que existan muestras entre el hyperplano y los vectores de soporte ($0 \leq \xi_\mu < 1$) o incluso en el lado opuesto del hiperplano ($\xi_\mu > 1$). El nuevo problema de optimización será entonces:

$$\theta(w, \xi) = \frac{||w||^2}{2} + C \sum_{\mu=1}^{|M|} \xi_\mu \tag{3.44}$$

con las restricciones:

$$l_\mu(w^T x_\mu + b) \geq 1 - \xi_\mu, \quad \xi_\mu \geq 0, \quad \forall \mu = 1, ..., |M| \tag{3.45}$$

El parámetro libre $C > 0$ ajusta el número de puntos a los que se les permite estar fuera de su correspondiente área. Un valor elevado de este parámetro implica un número bajo de errores permitidos.

3.4.2. SVMs con entrada difusa y salida difusa

Esta sección describe la extensión de los SVM clásicos a la implementación más reciente F^{2}SVM, introducida por primera vez en Thiel et al. (2007). Los valores de pertenencia m_μ^+ y m_μ^- para las clases positiva y negativa respectivamente deben ser definidos e incluidos en el problema de optimización:

$$\theta(w, \xi^+, \xi^-) = \frac{||w||^2}{2} + C \sum_{\mu=1}^{|M|} (\xi_\mu^+ m_\mu^+ + \xi_\mu^- m_\mu^-) \tag{3.46}$$

con las nuevas restricciones:

$$w^T x_\mu + b \geq 1 - \xi_\mu^+, \text{ with } \xi_\mu^+ \geq 0, \quad \forall \mu = 1, ..., |w| \tag{3.47}$$

$$w^T x_\mu + b \geq -1 + \xi_\mu^-, \text{ with } \xi_\mu^- \geq 0, \quad \forall \mu = 1, ..., |w| \tag{3.48}$$

Las nuevas cuatro restricciones también requieren que se introduzcan los multiplicadores de Lagrante $\alpha^+, \alpha^-, \beta^+ \beta^-$:

$$\begin{aligned} L(w, b, \xi^+, \xi^-, \alpha^+, \alpha^-, \beta^+, \beta^-) &= \frac{||w||^2}{2} + C \sum_{\mu=1}^{|M|} (\xi_\mu^+ m_\mu^+ + \xi_\mu^- m_\mu^-) \\ &- \sum_{\mu=1}^{|M|} \alpha_\mu^+ ((w^T x_\mu + b) - 1 + \xi_\mu^+) \\ &+ \sum_{\mu=1}^{|M|} \alpha_\mu^- ((w^T x_\mu + b) + 1 - \xi_\mu^-) \\ &- \sum_{\mu=1}^{|M|} \beta_\mu^+ \xi_\mu^+ - \sum_{\mu=1}^{|M|} \beta_\mu^- \xi_\mu^- \end{aligned} \tag{3.49}$$

Expresiones similares a 3.35 y 3.36 pueden ser obtenidas incluyendo las restricciones $\frac{\partial L}{\partial w} = 0$, $\frac{\partial L}{\partial b} = 0$, $\frac{\partial L}{\partial \xi_\mu^+} = 0$ y $\frac{\partial L}{\partial \xi_{mu}^-} = 0$:

$$\tilde{L}(\alpha) = \sum_{\mu=1}^{|M|} \alpha_\mu^+ + \sum_{\mu=1}^{|M|} \alpha_\mu^- - \frac{1}{2} \sum_{\nu=1}^{|M|} \sum_{\mu=1}^{|M|} (\alpha_\nu^+ - \alpha_\nu^-)(\alpha_\mu^+ - \alpha_\mu^-) x_\nu^T x_\mu \tag{3.50}$$

teniendo que $\alpha_\mu^+, \alpha_\mu^- > 0, \quad \forall \mu = 1, ..., |M|$ y sujeto a:

$$\sum_{\mu=1}^{|M|} (\alpha_\nu^+ - \alpha_\nu^-) = 0 \tag{3.51}$$

$$0 \leq \alpha_\mu^+ \leq C m_\mu^+, 0 \leq \alpha_\mu^- \leq C m_\mu^- \tag{3.52}$$

A continuación las condicioens de Karush-Kuhn-Tucker pueden ser obtenidas:

$$\alpha_\mu^+ ((w^T x_\mu + b) - 1 + \xi_\mu^+) = 0 \tag{3.53}$$

$$\alpha_\mu^- ((w^T x_\mu + b) + 1 - \xi_\mu^-) = 0 \tag{3.54}$$

Finalmente, aquellas muestras x_μ que verifican $(\alpha_\mu^+ - \alpha_\mu^-) \neq 0$ representan los vectores de soporte y definen la función de decisión:

$$y(x) = sign \left(\sum_{\mu=1}^{|M|} (\alpha_\mu^+ - \alpha_\mu^-) x^T x_\mu + b \right) \tag{3.55}$$

Para la extensión multi clase con un conjunto de muestras

$$M = \{(x_\mu, l_\mu) \mid x_\mu \in \mathbb{R}^n, l_\mu \in \mathbb{R}^k, with \sum_{j=1}^{k} l_{\mu,k} = 1, \quad \forall \mu = 1, ..., |M|\} \tag{3.56}$$

donde l_μ representa la etiqueta difusa o la probabilidad de pertenecer a cada una de las k clases. Con las extensiones descritas, el problema de clasificación rígida es meramente un caso particular del caso general F^{2}SVM, donde $l_{\mu,j} = 0$ para todas las clases excepto para una ($l_{\mu,j*} = 1$). Igualmente, para resolver el problema multi clase, $\frac{k(k-1)}{2}$ F^{2}SVMs en la configuración uno-contra-uno han de ser entrenados para cada par de clases i y j. Para entrenar cada uno de los F^{2}SVM, el conjunto de muestras es definido como:

$$M_{i,j} = \{(x_\mu, m^+_{\mu,i}) \mid m^+_{\mu,i} = l_{\mu,i}\} \cup \{(x_\mu, m^-_{\mu,j}) \mid m^-_{\mu,j} = l_{\mu,j}\} \tag{3.57}$$

Una vez que todos los F^{2}SVMs están entrenados, los pasos para generar una salida difusa son lo siguientes:

1. Para cada nueva muestra $z \in \mathbb{R}^n$ y F^{2}SVM entrenado $S_{i,j} \quad \forall i, j = 1, ..., k \mid i \neq j$.
2. Calcular la distancia $d_{i,j}(z) \in \mathbb{R}$ hasta el hiperplano correspondiente a $S_{i,j}$.
3. Transformar las distancias $d_{i,j}(z)$ usando la función de fermi $r_{i,j}(d_{i,j}) = \frac{1}{1+exp(-Ad_{i,j})}$, con A sujeto a optimización si requerido. Una representación de la función de fermi puede ser observada en la Figura 3.16.
4. La salida difusa $\tilde{y}(z)$ para todas las clases es obtenida como en Thiel et al. (2009) y Thiel (2009):
   I. Estimación de p_i mediante la media: $\hat{p}_i = \frac{\sum_{j \neq i} r_{ij}}{\frac{1}{2}k(k-1)}$.
   II. Actualización de las estimaciones de cada pareja: $\hat{\mu}_{ij} = \frac{\hat{p}_i}{\hat{p}_i + \hat{p}_j}$.
   III. Corregir los estimadores de la probabilidad de cada clase: $\hat{p}_i = \hat{p}_i \frac{\sum_{j \neq i} r_{ij}}{\sum_{j \neq i} \hat{\mu}_{ij}}$.
   IV. Normalizar las probabilidades de cada clase: $\hat{p}_i = \frac{\hat{p}_i}{\sum_{\forall j} \hat{p}_j}$.
   V. Iterar hasta la convergencia: Si el cambio en $\hat{p}_i > umbral$ vuelta a II.
5. Estimación de las probabilidades de cada clase normalizadas: $\tilde{y}(z) = (\hat{p}_1, ..., \hat{p}_k)$.

Los pasos seguidos para obtener $\tilde{y}(z)$ están representados en la Figura 3.17.

3.5. Fusión de los datos

Tal y como se explicón en la Sección 3.4, los SVMs son entrenados en una configuración uno-contra-uno. Esto significa que para cada uno de los 8 conjuntos de rasgos, un número de 15 SVMs han de ser entrenados, alcanzando un total de 120 evaluaciones (15 x 8) para cada nueva muestra. Sin embargo, todavía es necesario definir una fusión que permita combinar las 8 salidas en una única. Existen muchas formas para definir una estrategia de fusión (Kuncheva, 2001; Kuncheva et al., 2001; Kuncheva, 2004). La técnica utilizada en nuestro caso está basada en una sencilla multiplicación de las salidas difusas obtenidas a

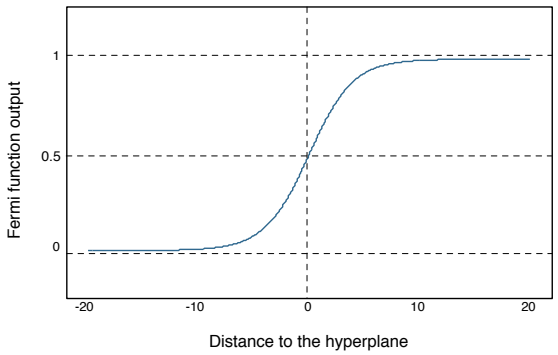


Figura 3.16: Función de Fermi utilizada para limitar la distancia $d_{i,j}(z)$ al rango $[0:1]$. Esta representación ha sido obtendia para un parámetro $A = 0{,}5$.

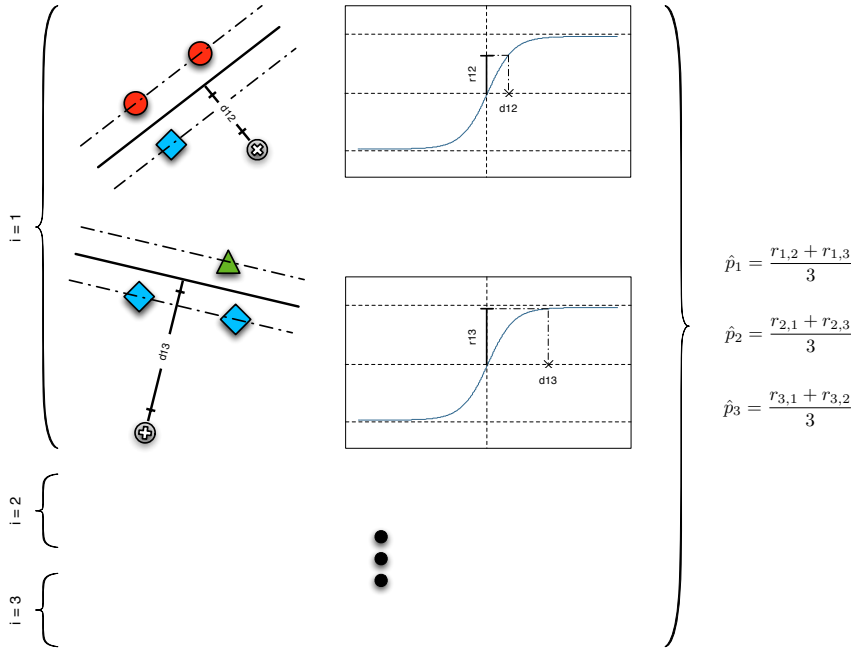
partir de cada conjunto de rasgos, seguida de una normalización. Durante los experimentos realizados se pudo observar que esta fusión tan sencilla era capaz de producir mejoras considerables sobre cada uno de los conjuntos de rasgos por separado, por lo que se decidió no profundizar más en implementaciones complejas. Esta aproximación está descrita en la Figura 3.18.

3.6. Aprendizaje Parcialmente Supervisado

A pesar de estar en posesión de todas las etiquetas para los datos, en esta parte de los experimentos asumiremos que no disponemos de todas, de modo que sea posible estudiar la precisión del sistema con respecto a la cantidad de trabajo invertida en etiquetar nuevas muestras.

3.6.1. Aprendizaje semi supervisado

El aprendizaje semi supervisado es normalmente utilizado cuando no todas las etiquetas están disponibles. En esta situación, las etiquetas que se poseen pueden ser usadas como una referencia para el sistema para automáticamente generar otras artificiales para el resto de los datos. Aunque existen diferentes técnicas que permiten realizar esta tarea, sólamente el algoritmo k-nearest neighbour (k-NN) será descrito y utilizado en este estudio. k-NN es un algoritmo de clasificación basado en las distancias a unas muestras de referencia. Distintos resultados pueden ser obtenidos mediante la modificación del parámetro k. Este parámetro controla la cantidad de muestras de entrenamiento utilizadas para emitir una decisión para cada nueva muestra sin etiqueta. En el caso más simple, 1-NN, sólamente sería necesario copiar la etiqueta de la muestra más cercana. En los experimentos realizados, sin embargo, un valor de $k = 5$ ha sido utilizado.



$$\hat{p}_1 = \frac{r_{1,2} + r_{1,3}}{3}$$

$$\hat{p}_2 = \frac{r_{2,1} + r_{2,3}}{3}$$

$$\hat{p}_3 = \frac{r_{3,1} + r_{3,2}}{3}$$

$$\begin{cases} \hat{\mu}_{1,2} = \dfrac{\hat{p}_1}{\hat{p}_1 + \hat{p}_2} \\ \hat{\mu}_{1,3} = \dfrac{\hat{p}_1}{\hat{p}_1 + \hat{p}_3} \end{cases} \longrightarrow \hat{p}_1 = \hat{p}_1 \frac{r_{1,2} + r_{1,3}}{\hat{\mu}_{1,2} + \hat{\mu}_{1,3}} \longrightarrow \hat{p}_1 = \frac{\hat{p}_1}{\hat{p}_1 + \hat{p}_2 + \hat{p}_3}$$

$$\begin{cases} \hat{\mu}_{3,1} = \dfrac{\hat{p}_3}{\hat{p}_1 + \hat{p}_3} \\ \hat{\mu}_{3,2} = \dfrac{\hat{p}_3}{\hat{p}_2 + \hat{p}_3} \end{cases} \longrightarrow \hat{p}_2 = \hat{p}_2 \frac{r_{2,1} + r_{2,3}}{\hat{\mu}_{2,1} + \hat{\mu}_{2,3}} \longrightarrow \hat{p}_2 = \frac{\hat{p}_2}{\hat{p}_1 + \hat{p}_2 + \hat{p}_3}$$

$$\begin{cases} \hat{\mu}_{2,1} = \dfrac{\hat{p}_2}{\hat{p}_2 + \hat{p}_1} \\ \hat{\mu}_{2,3} = \dfrac{\hat{p}_2}{\hat{p}_2 + \hat{p}_3} \end{cases} \longrightarrow \hat{p}_3 = \hat{p}_3 \frac{r_{3,1} + r_{3,2}}{\hat{\mu}_{3,1} + \hat{\mu}_{3,2}} \longrightarrow \hat{p}_3 = \frac{\hat{p}_3}{\hat{p}_1 + \hat{p}_2 + \hat{p}_3}$$

$$\tilde{y}(z) = (\hat{p}_1, \hat{p}_2, \hat{p}_3)$$

Figura 3.17: Generación de una salida difusa para un problema de tres clases. Este proceso se realiza para cada conjunto de rasgos por serparado, antes de que la fusión sea considerada. Primeramente, las distancias entre una nueva muestra y los hiperplanos son calculadas. Las distancias se transforman al rango $[0:1]$ mediante el uso de una función de Fermi. Con estos valores, las normalizaciones y correcciones descritas en los pasos I-IV son llevadas a cabo. Finalmente, la salida difusa para el conjunto de rasgos dado se obtiene agrupando las probabilidades normalizadas de las 3 clases.

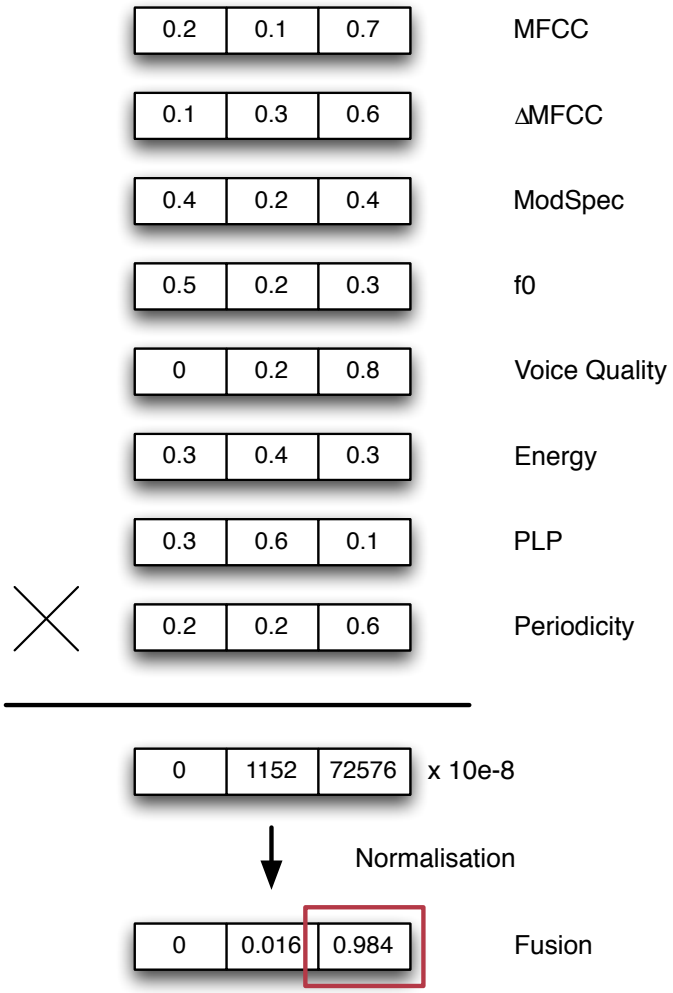


Figura 3.18: Ejemplo de fusión para las salidas difusas de los clasificadores. Para cada conjunto de rasgos existe un clasificador que produce una salida difusa. Las salidas de todos los clasificadores son combinadas para emitir una única salida mediante la definición de una fusión entre ellas. En esta figura, un problema de tres clases con los 8 conjuntos de rasgos descritos en la Sección 3.1 es representado. A partir de cada clasificador, una probabilidad de pertenencia a las 3 clases es obtenida, tal y como se muestra en la Figura 3.17. Las salidas de cada clasificador son multiplicadas y normalizadas, obteniendo un único vector que suma 1. La clase que se corresponde con la mayor de las probabilidades es la que será la decisión final del sistema.

Antes de describir los pasos para el auto aprendizaje, una descripción de las nuevas etiquetas que se desean obtener es necesaria. Puesto que un total de $J = 6$ clases son consideradas para nuestros experimentos, las etiquetas difusas han de contener 6 campos diferentes, p_j que representen el grado de pertenencia a cada una de las J clases:

$$l = \{p_j\}, \quad \forall j \in 1, ..., J \tag{3.58}$$

donde l_n representa la etiqueta difusa de la n-ésima muestra, y está sujeta a la restricción

$$\sum_{j=1}^{J} p_j = 1 \tag{3.59}$$

El primer paso para la creación de nuevas etiquetas difusas para las muestras desconocidas es el re-etiquetado del set de referencia. Éste, es un proceso sencillo puesto que la etiqueta real está disponible:

$$p_j = \begin{cases} 1 & : j = r \\ 0 & : j \neq r \end{cases}$$

donde $r \in \{1, ..., N\}$ representa la clase real a la que la muestra pertenece. Una vez que este paso ha concluido, un proceso iterativo de etiquetado puede ser llevado a cabo. Se considera iterativo puesto que cada nueva etiqueta que es generada pasa a considerarse una muestra de referencia para las sucesivas muestras.

Cuando una nueva muestra es considerada, la distancia euclídea a todos los puntos de referencia es calculada. De todas las distancias calculadas, sólamente las k más cercanas son consideradas. Con las etiquetas de estas k muestras l_n, la nueva etiqueta se puede calcular como:

$$l = \frac{1}{k} \sum_{n=1}^{k} l_n \tag{3.60}$$

La etiqueta generada es incluida en el set de referencia y considerada como correcta para todas las muestras futuras. Las iteraciones son repetidas para todas las muestras sin etiqueta. Un resumen de todo el proceso se muestra a continuación:

1. Extensión de las etiquetas de referencia a difusas.
2. Para cada muestra sin etiqueta:
    I. Calcular la distancia a todos los puntos en el set de referencia.

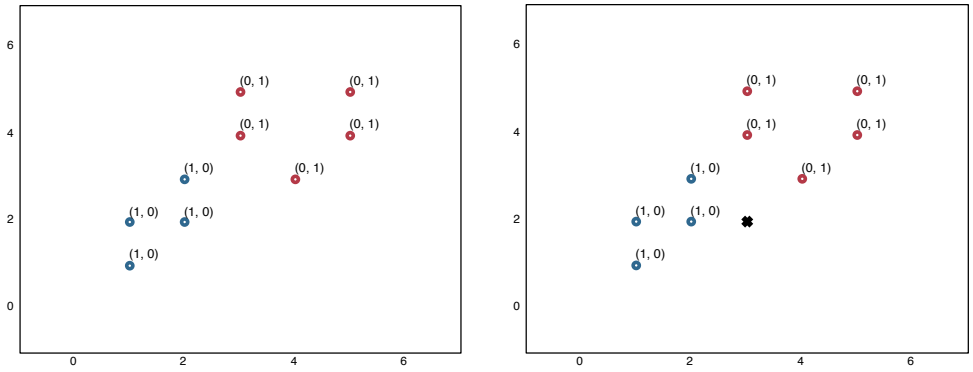


Figura 3.19: A la izquierda: Puntos del conjunto de referencia para dos clases diferentes (en azul y rojo respectivamente) con sus etiquetas reales extendidas a una representación difusa. A la derecha: Puntos de referencia y una observación (en negro) sin etiqueta.

II. Conservar la etiqueta solamente de las k muestras más cercanas.

III. Crear la nueva etiqueta difusa como la media de las citadas k etiquetas:

$$l = \frac{1}{k} \sum_{n=1}^{k} l_n$$

IV. Incluir la nueva etiqueta en el set de referencia.

Una representación del algoritmo se puede observar en las Figuras 3.19, 3.20 y 3.21. La evolución de los distintos sets descritos en este proceso conforme avanzan las iteraciones también está mostrada en la Figura 3.22.

Una vez que las etiquetas artificiales ha sido generadas, es necesario definir una medida de confianza que aporte información sobre el nivel de corrección que tienen las etiquetas. Puesto que no hay ningún experto supervisando el proceso, una cierta cantidad de error será introducida en las etiquetas inevitablemente. La medida de confianza permitirá discriminar las etiquetas que contienen demasiado error para evitar una penalización excesiva durante el entrenamiento. La medida propuesta para usar en estos experimentos es el valor que representa el mayor grado de pertenenecia a alguna de las clases. A partir de la definición de etiqueta difusa dada en la Ec. 3.58, la confianza de la etiqueta l puede ser obtenida como:

$$c = \max_j \{p_j\} \quad \forall j \in 1, ..., J \tag{3.61}$$

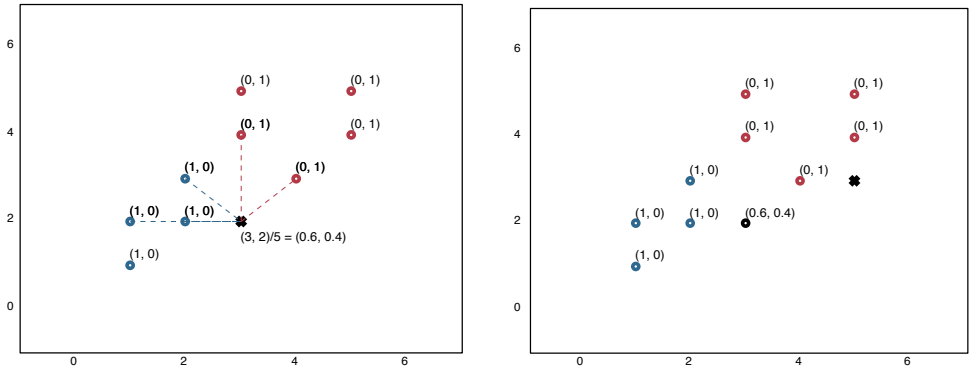


Figura 3.20: A la izquierda: distancia a los $k = 5$ vecinos más cercanos (nearest neighbors) y nueva etiqueta difusa para la nueva muestra. A la derecha: la nueva muestra y su etiqueta son incluídas en el set de entrenamiento. Una segunda observación sin etiqueta (cruz negra) llega al sistema.

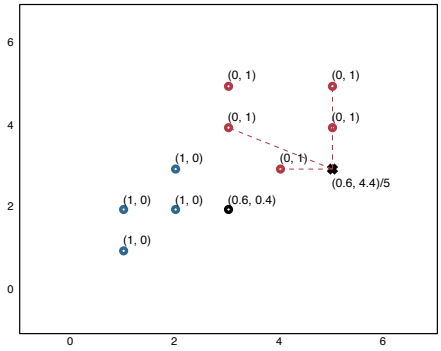


Figura 3.21: Distancias 5-NN para la nueva muestra y su nueva etiqueta difusa obtenida. La etiqueta k-NN de las muestras anteriores también se considera como una referencia para la iteración actual.

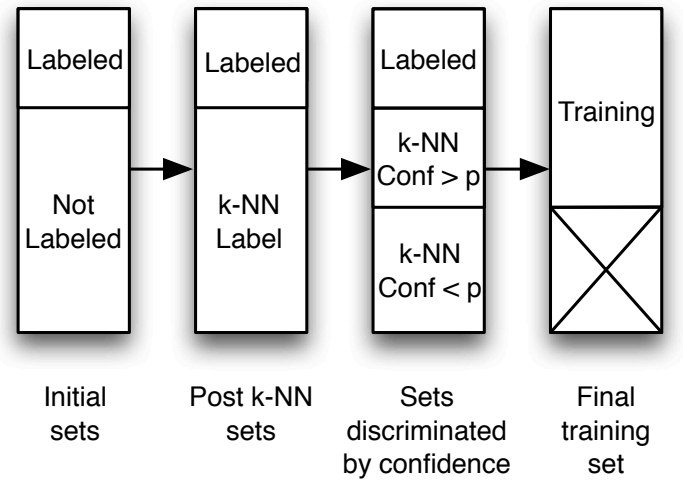


Figura 3.22: Evolución de los conjuntos de entrenamiento en el caso de entrenamiento semi-supervisado. Los conjuntos iniciales están formados por muestras etiquetadas y no etiquetadas. Mediante k-NN, las etiquetas desconocidas son generadas automáticamente. Aquellas etiquetas que presentan una confianza superior a un parámetro de discriminación $p \in [0, 1]$ son utilizadas para entrenamiento, mientras que el resto son descartadas.

donde p_j representa el grado de pertenencia de la muestra a la clase j. Un diagrama de flujo de la configuración de todo el experimento desde el etiqutado automático hasta la evaluación, es representando en la Figura 3.23.

Un parámetro de discriminación p es incluído en el sistema para decidir cuáles de las etiquetas automáticas se consideran válidas y cuáles no, basándose en la medida de confianza. Un valor alto de p (dentro del rango $[0, 1]$) implica el uso de etiquetas con una alta confianza, mientras que se reducirá la cantidad de ellas que pueden ser utilizadas para entrenamiento.

3.6.2. Aprendizaje activo

Las técnicas tradicionales de aprendizaje dependen en general de una gran cantidad de muestras etiquetadas, distribuidas sobre el espacio de rasgos, con tanta información como sea posible acerca de la distribución que las ha generado. A mayor cantidad de datos, mayor es la probabilidad de que el sistema sea capaz de aprender las características de la distribución (suponiendo que las hay) y emitir decisiones correctas para nuevas muestras. Sin embargo, este tipo de escenarios puede tener algunos inconvenientes, como la dificultad de etiquetar datos (puede ser costoso para muchas aplicaciones) que puede conducir a

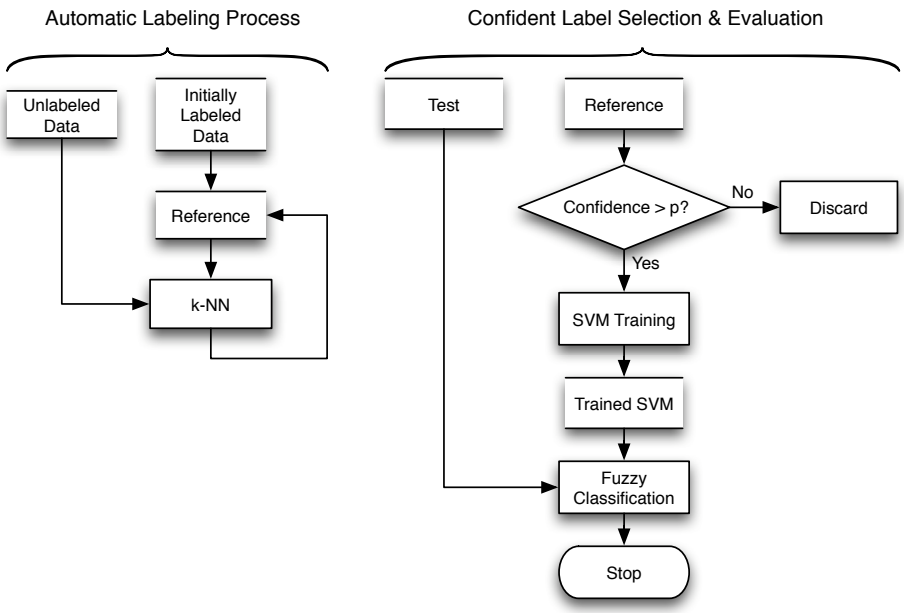


Figura 3.23: Diagrama de flujo que describe la solución para el aprendizaje semi-supervisado. A la izquierda: proceso de etiquetado automático; usando un conjunto de referencia etiquetado, nuevas etiquetas pueden ser generadas para las muestras que no tienen, mediante el algoritmo k-NN. Una vez generadas son incluidas en el conjunto de referencia. A la derecha: selección de muestras con una alta confianza y evaluación; un proceso de discriminación es llevado a cabo para decidir qué etiquetas k-NN son suficientemente buenas como para ser utilizadas en el entrenamiento de los SVMs.

situaciones en las que no se disponga de suficientes datos etiquetados. Si fuese posible, por ejemplo, cambiar la suposición .ª mayor cantidad de muestras etiquetadas disponibles, mejor se puede entrenar" por çuando mejor sean nuestras muestras etiquetadas, mejor se puede entrenar" entonces sería posible conseguir buenos entrenamientos con cantidades de datos más reducidas. Ésta es precisamente la idea que hay detrás del aprendizaje activo. En un conjunto de entreanmiento elegido aleatoriamente, la probabilidad de muestras poco representativas o redundantes es bastante alta. Por tanto, el sistema de aprendizaje puede estar desperdiciando parte del esfuerzo y coste puesto en el etiquetado. Esto es así debido a que no es sencillo saber si una muestra es más o menos representativa a priori.

Sin embargo, si al sistema se le permite solicitar las etiquetas de aquellas muestras que quiere utilizar para ser entrenado, es probable que elija las más útiles, basándose en la proximidad a las fronteras de decisión entre las distintas cla-

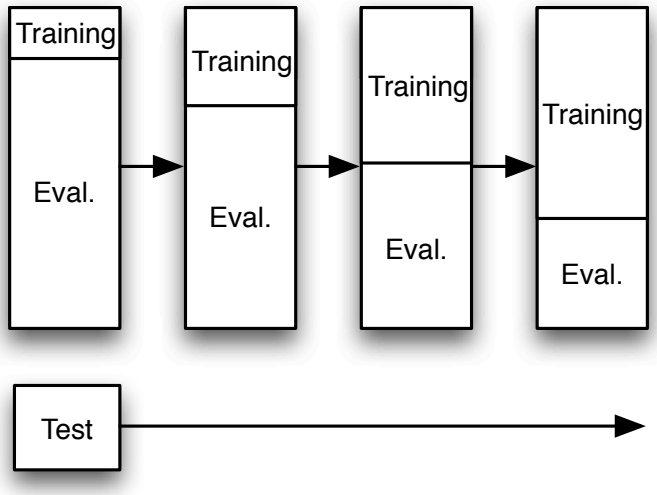


Figura 3.24: Evolución de los distintos conjuntos de entrenamiento en el caso de aprendizaje activo. El conjunto de evaluación está formado por una serie de muestras de entre las cuales el sistema decide en cada iteración las que quiere que sean etiquetadas. Las muestras que obtienen una etiqueta por parte de un experto pasan a formar parte del conjunto de entrenamiento, mientras que el conjunto de test se mantiene intacto durante todo el proceso.

ses. Hay distintos formas de interpretar esta idea, y un buen resumen de éstas puede encontrarse en Settles (2009). La técnica empleada para este estudio es la descrita en esta sección.

Inicialmente, el conjunto de todas las muestras con etiquetas es dividido en dos grupos: entrenamiento y evaluación. El conjunto de evaluación no cambia durante todo el proceso. El de entrenamiento, sin embargo, representa un conjunto de muestras disponibles de entre las cuales el sistema decidirá en cada iteración qué etiquetas quiere conocer y utilizar para entrenarse. La evolución de los distintos conjuntos conforme avanzan las iteraciones se muestra en la Figura 3.24.

Un pequeño número de etiquetas es utilizado inicialmente para el entrenamiento. Entonces se intenta clasificar todas las muestras no utilizadas. Para cada una de estas muestras, el clasificador produce una salida difusa que representa el grado de pertenencia a cada una de las distintas clases, siendo la suma de todos ellos igual a 1. Por tanto, considerando el valor más alto de pertenencia a una clase se obtiene la clase más probable. Es posible entonces definir el nivel de confianza de la muestra como el grado de pertenencia a la clase más probable,

tal y como se definió en la Ec. 3.61. Considerar solamente la clase más probable para cada muestra proporciona una medida de la confianza de la misma. Bajo esta suposición, tiene sentido pensar que aquellas muestras con una baja confianza son aquellas que el sistema tiene problemas para clasificar. Si esas muestras son correctamente etiquetadas por un experto y entonces utilizadas para el entrenamiento del sistema, éste será capaz de aprender las características de zonas en el espacio de los datos donde no tiene mucha confianza. Estas áreas representan, en general, los bordes de las áreas de decisión, donde el solapamiento de clases es frecuente y una mayor cantidad de información es requerida. Por otro lado, para aquellas muestras que presentan una buena confianza, el sistema no requiere más información puesto que representan una tarea fácil para él. Un diagrama de flujo del aprendizaje activo y su evaluación está representado en la Figura 3.25.

Un ejemplo gráfico del proceso de aprendizaje activo se muestra en la Figura 3.26.

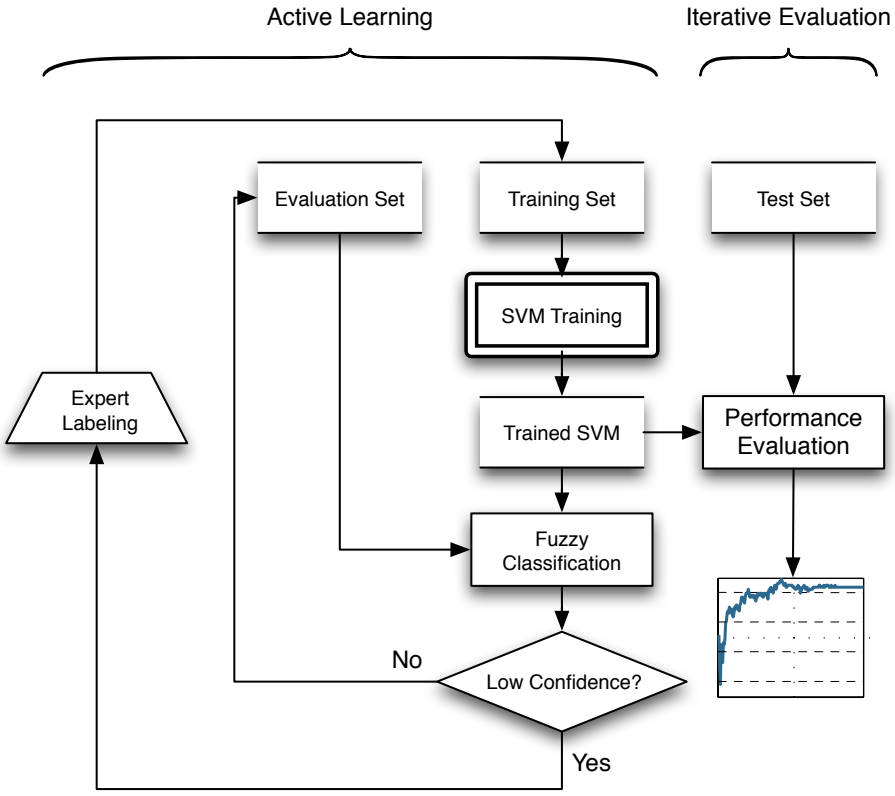


Figura 3.25: Diagrama de flujo de la solución propuesta para aprendizaje activo, incluyendo las partes de entrenamiento y evaluación iterativa. A la izquierda: las SVMs son entrenadas inicialmente con un conjunto reducido de muestras. Un conjunto de validación es utilizado como entrada al clasificador y la confianza de la decisión es calculada. Aquellas muestras que producen una baja confianza son etiquetadas y usadas para entrenamiento. A la derecha: proceso de evaluación, llevado a cabo en cada iteración con un mismo conjunto de test.

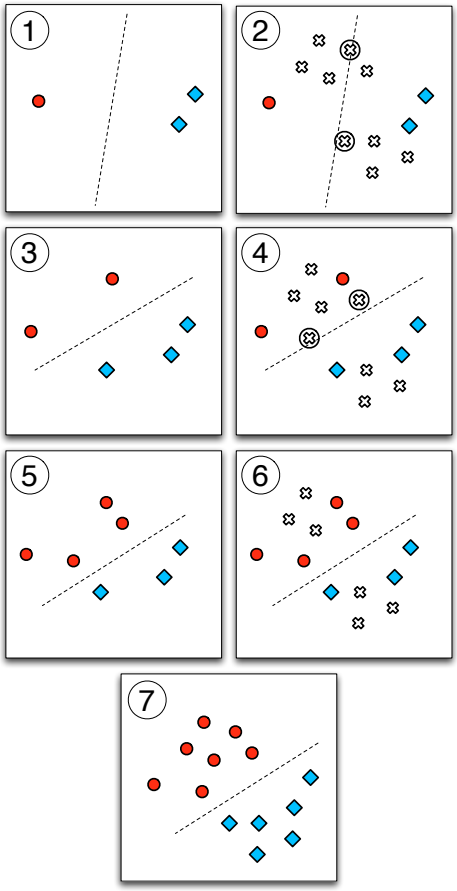


Figura 3.26: Proceso de aprendizaje activo. Datos de dos clases (en azul y rojo) son considerados. En un primer paso, 3 muestras aleatorias son etiquetadas y un clasificador es entrenado con ellas ①. El conjunto de datos sin etiqueta es evaluado con este clasificador y las muestras con menor confianza (aquellas más cercanas a las fronteras de decisión) son seleccionadas ②. Un experto etiqueta las muestras seleccionadas y entonces se convierten en parte del conjunto de entrenamiento, definiendo un nuevo clasificador ③. El conjunto restante de muestras sin etiqueta, una vez más, se pasa por el clasificador y se eligen nuevas muestras para ser etiquetadas ④. Estos pasos se repiten ⑤, ⑥ hasta que una precisión suficientemente buena sea alcanzada. Las etiquetas de todas las muestras son mostradas en ⑦. Sin embargo, no todas ellas habrían sido necesarias para construir un buen clasificador. El definido en el paso ④ parece lo suficientemente bueno, por lo que no habría sido necesario continuar con el proceso en este caso particular.

4 Experimentos y Resultados

Los experimentos realizados en este estudio han sido diseñados atendiendo a tres problemas diferentes. Primeramente a la necesidad de estudiar las emociones humanas y sus características basadas en la voz. Para evaluación del sistema propuesto, un modelo es entrenado en un estilo supervisado. Con este modelo, tests de reconocimiento automatico de emociones han sido realizados. En una segunda parte de los experimentos, surge la discusión sobre el efecto del aprendizaje parcialmente supervisado en nuestros resultados. Con ese propósito, la generación automática de etiquetas (para el caso de aprendizaje semi-supervisado) es estudiada, al igual que aprendizaje activo.

Para evaluación de los resultados, matrices de confusión han sido generadas con los resultados obtenidos para la clasificación automática y comparadas con las obtenidas por los humanos en los tests de percepción. En las matrices, cada fila suma uno, mostrando la cantidad de muestras de cada clase que son reconocidas por el sistema como cada una de las posibles clases. Las columnas, que no necesariamente suman uno, muestran qué cantidad de muestras de cada clase es reconocida como perteneciente a una particular. La diagonal principal de las matrices de confusión representa, por tanto, la precisión media para cada clase (el porcentaje de muestras de una clase dada que son correctamente clasificadas). La precisión media sobre todas las clases es calculada como el valor medio de la diagonal principal. En un caso en el que la clasificación fuese perfecta, el resultado obtenido sería la matriz identidad.

La arquitectura del sistema utilizado se muestra en la Figura 4.1.

4.1. Aprendizaje Supervisado

En esta parte de los experimentos, etiquetas rígidas para cada una de las muestras de voz están disponibles para el entrenamiento. Estas etiquetas son utilizadas para entrenar los SVMs tal y como se explica en la Sección 3.4 en una configuración entrada-rígida salida-rígida.

Los distintos rasgos son extraídos y alineados para todas las muestras disponibles, tal y como se describió en la Sección 3.1, contando con un total de 8 sets

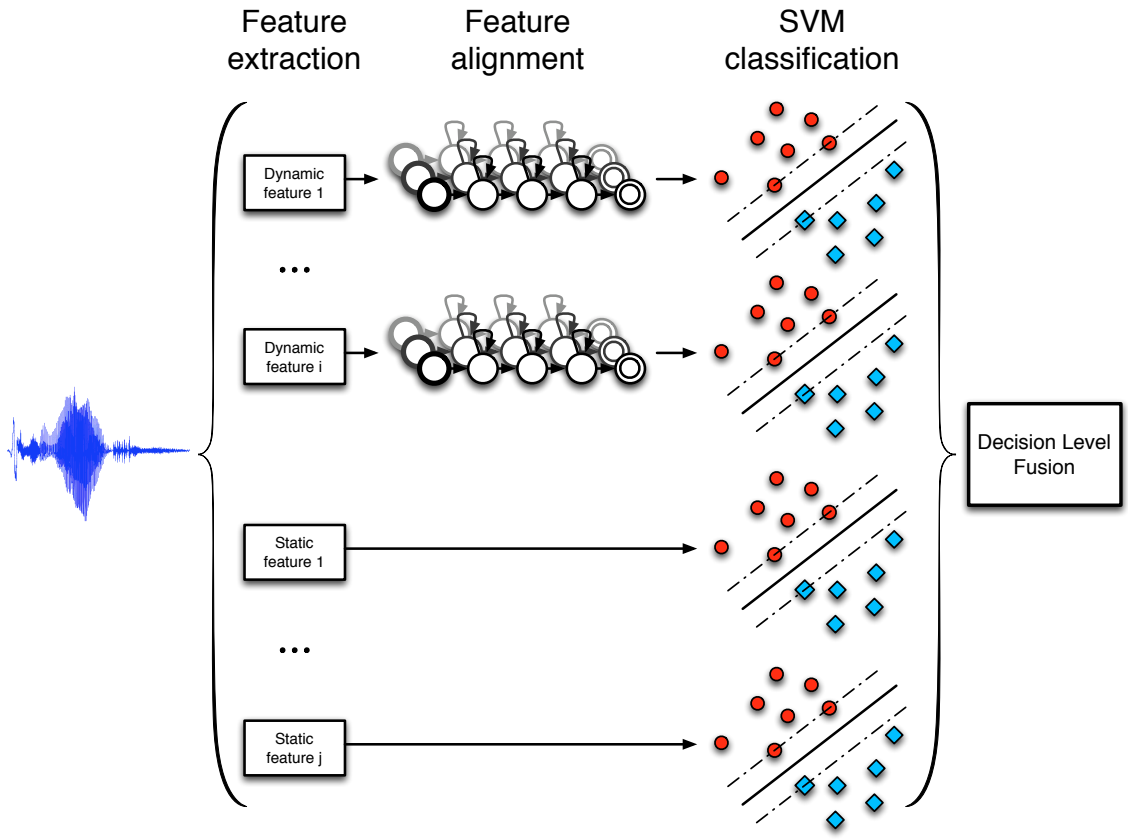


Figura 4.1: Arquitectura del sistema propuesto. Los rasgos han de sufrir un proceso de alineamiento basado en la verosimilitud de los HMM antes de ser utilizados para entrenamiento o test, tal y como se describe en la Sección 3.2.3. Los rasgos estáticos no requieren el proceso de alineamiento puesto que ya tienen una longitud fija. Tras el alineamiento, la clasificación en los distintos SVMs es llevada a cabo y, finalmente, decisiones son tomadas en base a la fusión definida para todos los conjuntos de rasgos.

Cuadro 4.1: Matriz de confusión para los experimentos de clasificación automática para el caso emociones actuadas por un hombre, conducidos con el conjunto WaSeP. El acierto medio es del 87 %.

	F	D	H	N	S	A
Fear	.87	.01	.06	.01	.00	.05
Disgust	.01	.92	.06	.00	.00	.01
Happiness	.13	.07	.67	.09	.01	.04
Neutral	.00	.01	.06	.93	.00	.00
Sadness	.00	.00	.00	.00	.99	.00
Anger	.01	.07	.04	.02	.00	.85

Cuadro 4.2: Matriz de confusión para los experimentos de clasificación automática para el caso emociones actuadas por una mujer, conducidos con el conjunto WaSeP. El acierto medio es del 84 %.

	F	D	H	N	S	A
Fear	.74	.04	.16	.01	.01	.05
Disgust	.02	.90	.04	.01	.01	.03
Happiness	.03	.02	.77	.13	.02	.03
Neutral	.00	.00	.09	.87	.04	.00
Sadness	.01	.00	.02	.05	.91	.00
Anger	.01	.07	.02	.02	.01	.88

Cuadro 4.3: Matriz de confusión para los experimentos de clasificación automática para el caso emociones actuadas por un hombre o mujer indistintamente, conducidos con el conjunto WaSeP. El acierto medio es del 84 %.

	F	D	H	N	S	A
Fear	.80	.03	.08	.01	.02	.06
Disgust	.01	.88	.05	.00	.04	.03
Happiness	.08	.02	.71	.12	.04	.03
Neutral	.00	.01	.16	.82	.01	.00
Sadness	.02	.00	.03	.01	.95	.00
Anger	.01	.07	.02	.03	.00	.86

para cada muestra. Con los rasgos alineados, dos sets son creados distribuidos al 90 % y 10 % para entrenamiento y evaluación de los SVMs respectivamente. Este proceso es repetido para realizar 10 veces realizando una validación cruzada.

Las matrices de confusión obtenidas con estos tests se muestran en los Cuadros 4.1 , 4.2 y 4.3 para el caso de hombre, mujer e independencia de género, con el conjunto de datos WaSeP.

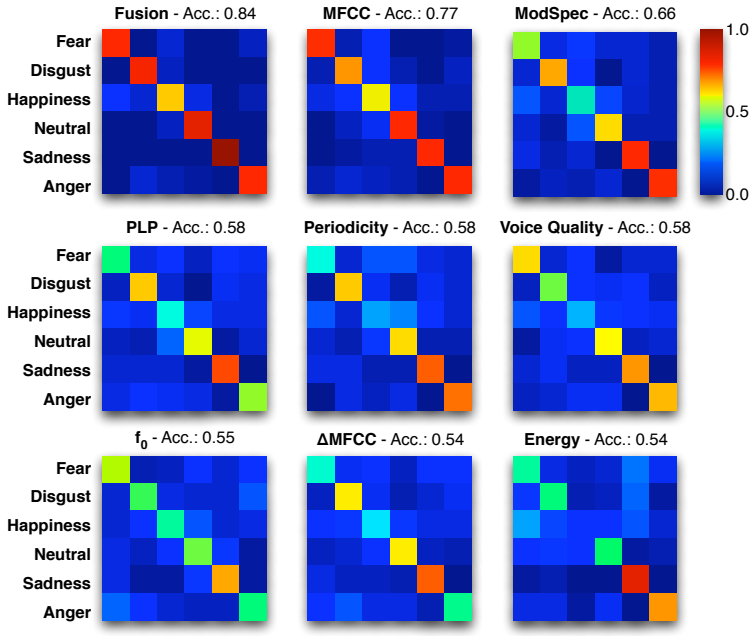


Figura 4.2: Matrices de confusión para los experimentos de clasificación automática con el corpus WaSeP. Las matrices están ordenadas por orden de precisión, comenzando con la fusión de todos los conjuntos de rasgos. Las columnas y filas están en el mismo orden, siendo la diagonal principal la precisión alcanzada para cada clase. Las filas suman 1, ya que representan el total de etiquetas consideradas de cada clase. Las columnas, sin embargo, no necesariamente suman 1, puesto que representan la salida del clasificador. Los colores cálidos y frios representan valores altos y bajos respectivamente, tal y como representa la barra de color a la derecha.

La precisión media obtenida en estos casos es de 87 % para el hombre, 85 % para la mujer y 84 % para el caso conjunto. En todos los experimentos, felicidad es la emoción con un menor acierto, no siendo en ningún caso superior al 80 %. Contrasta con el caso de tristeza, que es, en todos los casos, clasificada con un acierto superior al 90 %. Para observar mejor los resultados de la fusión, se ha generado la Figura 4.2. Esta figura presenta imágenes escaladas de las matrices de confusión producidas por cada uno de los conjuntos de rasgos por separado, así como de su fusión.

Para comparar los resultados, los experimentos también han sido conducidos sobre el conjunto EmoDB. La evaluación del sistema en este caso muestra una precisión del 77 %, ligeramente superior al obtenido con WaSeP. La matriz de confusión para este test puede observarse en el Cuadro 4.4 y su correspondiente imagen escalada se muestra en la Figura 4.3.

Cuadro 4.4: Matriz de confusión para los experimentos de clasificación automática, para el caso de independencia de género, conducidos con el conjunto EmoDB.

	F	D	H	N	S	A	B
Fear	.77	.01	.10	.01	.06	.00	.05
Disgust	.10	.69	.04	.04	.07	.01	.05
Happiness	.08	.02	.53	.00	.03	.00	.33
Neutral	.00	.01	.00	.80	.16	.03	.00
Sadness	.01	.01	.00	.06	.92	.00	.00
Anger	.00	.00	.00	.03	.07	.89	.00
Boredom	.04	.01	.14	.00	.00	.00	.81

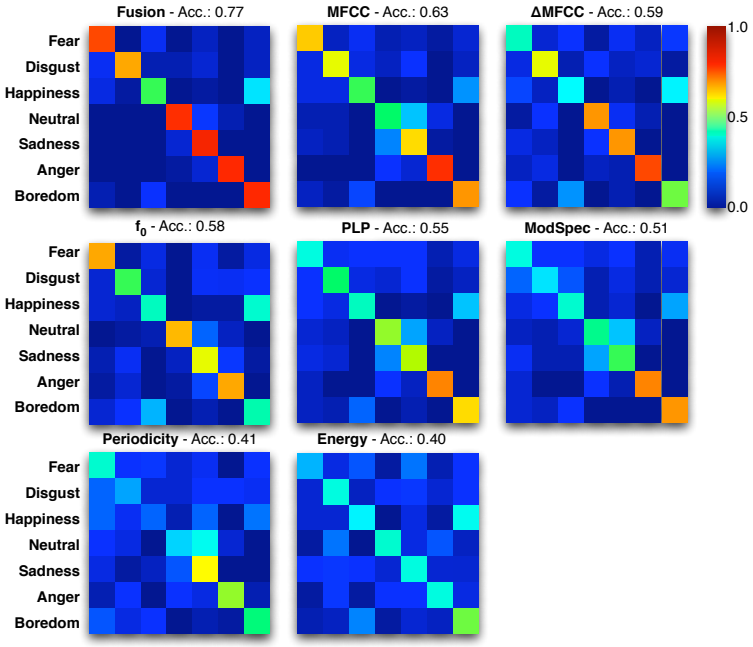


Figura 4.3: Matrices de confusión para los experimentos de clasificación automática con el corpus EmoDB. Las matrices están ordenadas por orden de precisión, comenzando con la fusión de todos los conjuntos de rasgos. Las columnas y filas están en el mismo orden, siendo la diagonal principal la precisión alcanzada para cada clase. Las filas suman 1, ya que representan el total de etiquetas consideradas de cada clase. Las columnas, sin embargo, no necesariamente suman 1, puesto que representan la salida del clasificador. Los colores cálidos y frios representan valores altos y bajos respectivamente, tal y como representa la barra de color a la derecha. En este caso el conjunto de rasgos Cualidad de Voz no son utilizados, puesto que su extracción para secuencias largas es compleja y anotación a nivel de fonema habría sido necesaria.

4.2. Aprendizaje Parcialmente Supervisado

También se han realizado experimentos dentro del marco del aprendizaje parcialmente supervisado. Primeramente, una aproximación semi-supervisada, donde el sistema es capaz de general etiquetas automáticamente para las muestras, a partir únicamente de unos puntos de referencia, etiquetados por un experto. En la siguiente estrategia estudiada (aprendizaje activo), es el experto quien etiqueta las muestras, pero en este caso únicamente aquellas que son elegidas por el propio sistema.

4.2.1. Aprendizaje semi-supervisado

En esta sección no solo se presentan resultados finales sino también parciales, para permitir una mejor comprensión de las técnicas descritas en la Sección 3.6 y su nivel de precisión para un valor del parámetro $k = 5$.

En un primer experimento, etiquetas difusas son generadas automáticamente para 600 muestras, siguiendo el procedimiento descrito en las Figuras 3.19, 3.20 y 3.21. Como set de referencia inicial, 50 muestras por cada clase son utilizadas. En la Figura 4.4 se muestran los conjuntos de referencia y de test. Para el último, tanto las etiquetas reales como las artificiales son mostradas. Cada color diferente representa una de las seis clases consideradas. Puesto que las etiquetas generadas mediante k-NN son difusas, sólamente la clase con un mayor grado de pertenencia es representada. No obstante, es posible observar la similitud obtenida con las etiquetas artificiales, comparadas con las reales. Para representar mejor la cantidad de información que contienen las etiquetas difusas, se ha generado la Figura 4.5. Ésta muestra la cantidad de etiquetas correctas y erróneas generadas mediante k-NN, considerando no sólo la clase con un mayor grado de pertenencia, sino también la segunda y tercera.

La relación de etiquetas correctas en las tres distintas situaciones es de 52 %, 77 % y 89 % para los casos de 1, 2 y 3 clases con mayor grado de pertenencia, respectivamente.

Como ha podido observarse, cuando existen suficientes puntos de referencia, k-NN produce resultados considerablemente buenos para generar etiquetas difusas automáticamente. Sin embargo, nuestros experimentos deben suponer una situación en la que el conjunto de referencia es mucho más reducido. Distintos experimentos han sido conducidos dentro de esta propuesta con la intención de producir una mejora significativa en la clasificación, cuando el sistema es entrenado con un conjunto reducido de etiquetas rígidas, extendido con las etiquetas difusas automáticamente generadas. La precisión de referencia (baseline) en este experimento ha sido reducida para simular una situación con una cantidad pequeña de datos disponible. Esta baseline proporciona una precisión media de 73 % para el caso de independencia de género con el conjunto WaSeP. Un barrido sobre el parámetro p muestra que el máximo se encuentra para un

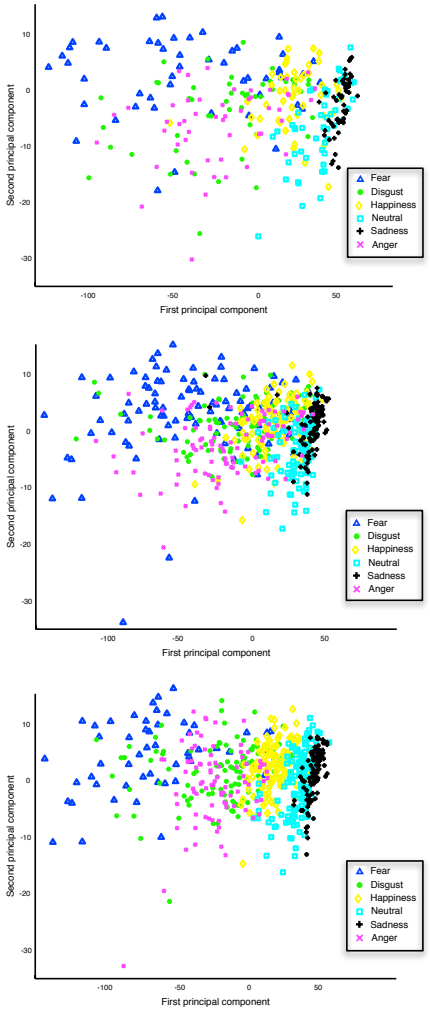


Figura 4.4: Las tres figuras corresponden a las dos componentes principales de MFCC, obtenidas a partir de la voz de una mujer en el corpus WaSeP. La primera figura representa los vectores de referencia, anotados por un experto (50 muestras por cada clase). La figura del centro representa las etiquetas reales del conjunto de test. La última figura muestra una representación de la etiqueta automáticamente obtenida mediante k-NN con un parámetro $k = 5$. En este último caso, puesto que se trata de etiquetas difusas, la clase mostrada es aquella con un mayor grado de pertenencia. Suponiendo que el etiquetado automático fuera perfecto, la segunda y tercera figuras deberían ser idénticas.

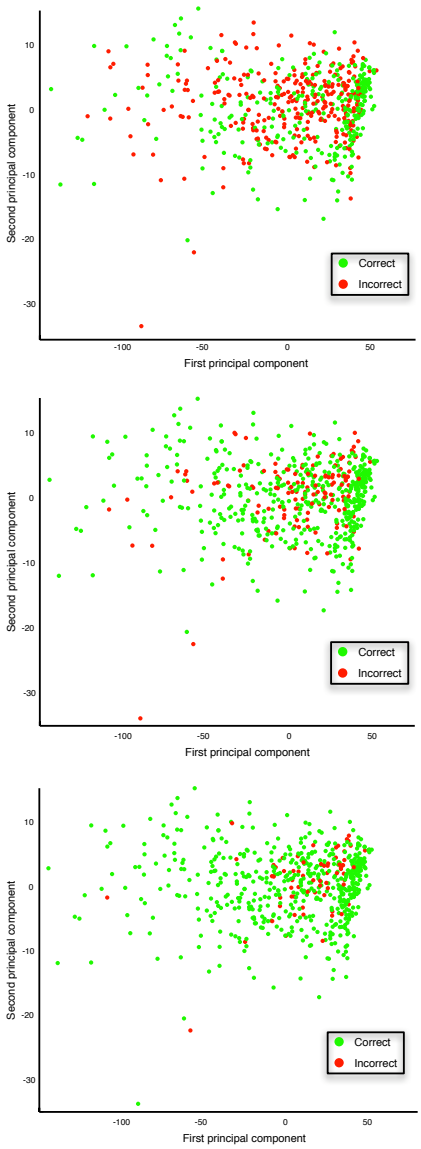


Figura 4.5: Representación de etiquetas correctas (verde) e incorrectas (rojo) generadas mediante k-NN con un parámetro $k = 5$, utilizando como conjunto de referencia de 50 muestras por cada clase. La primera figura muestra las etiquetas correctas, considerando únicamente la clase que presenta un mayor grado de pertenencia para cada una de ellas. La figura central, no sólo considera aquella clase con un mayor grado de pertenencia, sino las dos mayores. La última figura incluye también la tercera clase.

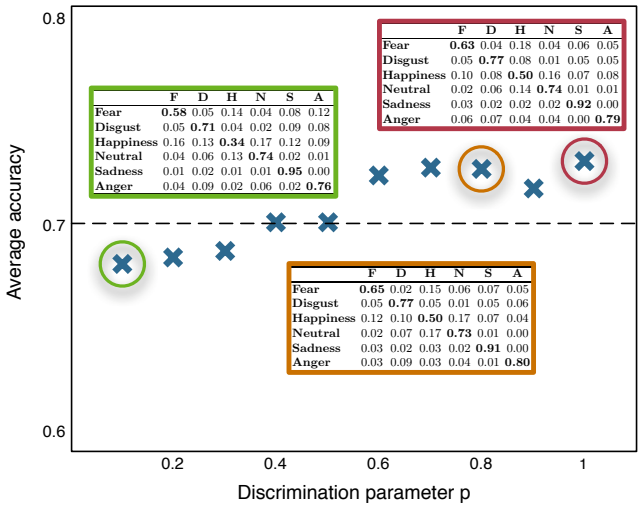


Figura 4.6: Precisión media obtenida para distintos valores del parámetro de discriminación p. También se muestran matrices de confusión para los valores de p 0.1, 0.8, así como la baseline, representada como p = 1.

Cuadro 4.5: Matriz de confusión de los experimentos con independencia de género realizados con el corpus WaSeP, para un parámetro de discriminación $p = 0{,}1$. La precisión media obtenida es del 68 %.

	F	D	H	N	S	A
Fear	.58	.05	.14	.04	.08	.12
Disgust	.05	.71	.04	.02	.09	.08
Happiness	.16	.13	.34	.17	.12	.09
Neutral	.04	.06	.13	.74	.02	.01
Sadness	.01	.02	.01	.01	.95	.00
Anger	.04	.09	.02	.06	.02	.76

valor $p = 0{,}8$, alcanzando una precisión media de 73 %. Una representación gráfica de este análisis se muestra en la Figura 4.6. Los resultados obtenidos se muestran en forma de matriz de confusión en los Cuadros 4.5, 4.6 y 4.7 para valores de discriminación p igual a 0,1, 0,8 y 1 respectivamente.

Otro experimento ha sido llevado a cabo para verificar el efecto del parámetro de discriminación p sobre la precisión cuando las etiquetas k-NN son generadas con un conjunto de referencia más grande. El entrenamiento de los SVMs es entonces realizado con las etiquetas k-NN junto con una parte reducida del conjunto de referencia, tal y como se describe en la Figura 4.7. De este modo, es posible observar el efecto el parámetro de discriminación p sobre la precisión del sistema, tal y como se muestra en la Figura 4.8.

Cuadro 4.6: Matriz de confusión de los experimentos con independencia de género realizados con el corpus WaSeP, para un parámetro de discriminación $p = 0{,}8$. La precisión media obtenida es del 73 %.

	F	D	H	N	S	A
Fear	.65	.02	.15	.06	.07	.05
Disgust	.05	.77	.05	.01	.05	.06
Happiness	.12	.10	.50	.17	.07	.04
Neutral	.02	.07	.17	.73	.01	.00
Sadness	.03	.02	.03	.02	.91	.00
Anger	.03	.09	.03	.04	.01	.80

Cuadro 4.7: Matriz de confusión de los experimentos con independencia de género realizados con el corpus WaSeP, para un parámetro de discriminación $p = 1$. Este caso coincide con la baseline, dado que no se considera ninguna etiqueta k-NN para entrenamiento, al no existir ninguna con una medida de confianza superior a 1. La precisión media obtenida es del 73 %.

	F	D	H	N	S	A
Fear	.63	.04	.18	.04	.06	.05
Disgust	.05	.77	.08	.01	.05	.05
Happiness	.10	.08	.50	.16	.07	.08
Neutral	.02	.06	.14	.74	.01	.01
Sadness	.03	.02	.02	.02	.92	.00
Anger	.06	.07	.04	.04	.00	.79

Además, se observa que es posible aumentar la precisión en un 10 % extendiendo el conjunto de entrenamiento con etiquetas generadas automáticamente. Esto prueba que las etiquetas artificiales pueden ser utilizadas para entrenar el sistema, siempre y cuando la cantidad de error introducida artificialmente se mantenga tan baja como sea posible.

4.2.2. Aprendizaje Activo

Estos experimentos han sido diseñados con la intención de evaluar la respuesta del sistema cuando el conjunto de entrenamiento es elegido por él mismo. El diagrama de flujo de la configuración del sistema en este caso está representada en la Figura 3.25. En el entrenamiento iterativo y proceso de test, cada iteración representa un aumento de 10 muestras en el conjunto de entrenamiento. Para evaluar los resultados obtenidos en esta sección, la Figura 4.9 ha sido generada. Esta figura muestra la corrección media del sistema para cada iteración, en los experimentos con independencia de genénero del conjunto WaSeP. Los Cuadros 4.8, 4.9 y 4.10 muestran las matrices de confusión para los resultados obtenidos con aprendizaje activo después de todas las iteraciones, es decir, con todas

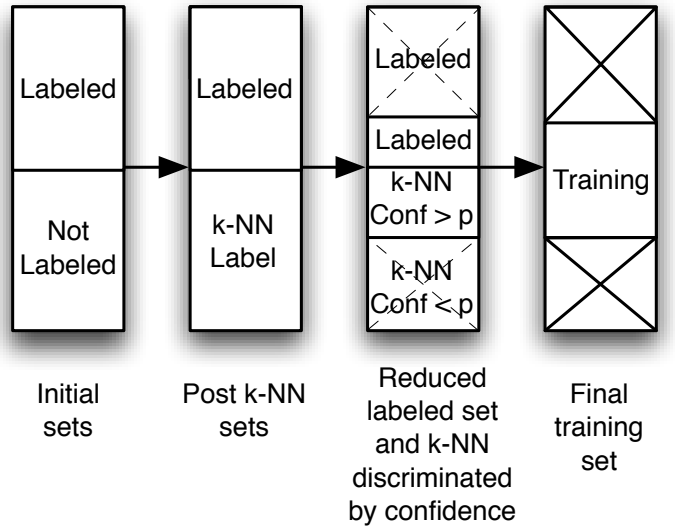


Figura 4.7: Evolución de los sets de entrenamiento en el experimento para validar la medida de confianza propuesta. Las etiquetas k-NN son generadas a partir de un set de referencia extendido para reducir el error incluido en ellas. A continuación, para entrenar los SVMs, no todo el set de referencia es utilizado, sino solamente una pequeña fracción de él, juto con las etiquetas artificiales con una confianza mayor que el parámetro de discriminación p.

las muestras disponibles etiquetadas y utilizadas para entrenar el sistema. La precisión media obtenida en este caso, 88 % considerando independencia de género, es superior al 84 % obtenido en la Sección 4.1 debido a que se ha utilizado un conjunto de entrenamiento más grande que pueda representar bien el efecto iterativo del entrenamiento activo. Es posible observar que un 80 % de precisión es alcanzado tras escasamente 40 iteraciones. Esto significa que tan solo usando un tercio de la cantidad de datos utilizados en los experimentos previos, un nivel de precisión similar es alcanzando. Aún más, debe observarse que tras ciertas iteraciones, un punto de saturación es alcanzado con solamente una pequeña parte de todos los datos disponibles utilizados.

4.3. Discusión

Las matrices de confusión obtenidas en la Sección 4.1 proporcionan una buena base para comparar los aciertos y errores humanos con los obtenidos artificialmente. Un primer vistazo sobre ellas muestra que los resultados obtenidos son bastante similares en general. Con el conjunto WaSeP, el 84 % de precisión ob-

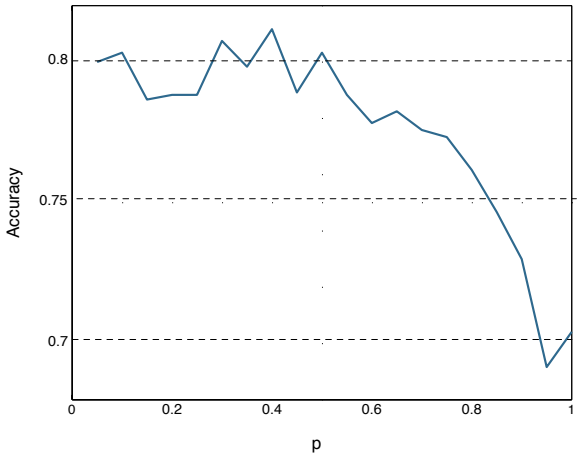


Figura 4.8: Corrección de la clasificación para distintos valores del parámetro p usando etiquetas k-NN generadas con un conjunto de referencia grande. Más tarde el conjunto de referencia fue reducido en un 90 %, utilizando el únicamente el 10 % restante junto con las nuevas etiquetas para entrenar los SVMs.

Cuadro 4.8: Matriz de confusión obtenida para los experimentos de aprendizaje activo llevados a cabo con voz de hombre del conjunto WaSeP. La precisión media alcanzada es del 90 %.

	F	D	H	N	S	A
Fear	.85	.01	.12	.00	.00	.02
Disgust	.00	.92	.07	.00	.00	.01
Happiness	.09	.02	.80	.06	.00	.03
Neutral	.00	.00	.06	.93	.00	.00
Sadness	.00	.00	.01	.00	.99	.00
Anger	.01	.02	.03	.01	.00	.93

Cuadro 4.9: Matriz de confusión obtenida para los experimentos de aprendizaje activo llevados a cabo con voz de mujer del conjunto WaSeP. La precisión media alcanzada es del 85 %.

	F	D	H	N	S	A
Fear	.72	.04	.17	.01	.01	.05
Disgust	.02	.86	.06	.01	.01	.04
Happiness	.02	.02	.82	.10	.02	.02
Neutral	.01	.00	.12	.85	.02	.00
Sadness	.01	.00	.02	.03	.94	.00
Anger	.01	.03	.02	.01	.00	.93

Cuadro 4.10: Matriz de confusión obtenida para los experimentos de aprendizaje activo llevados a cabo con voz de hombre del conjunto WaSeP. La precisión media alcanzada es del 88 %.

	F	D	H	N	S	A
Fear	.83	.01	.11	.01	.01	.02
Disgust	.01	.89	.05	.01	.02	.03
Happiness	.05	.02	.79	.10	.02	.02
Neutral	.01	.00	.10	.88	.00	.01
Sadness	.01	.00	.02	.01	.97	.00
Anger	.01	.02	.02	.01	.00	.93

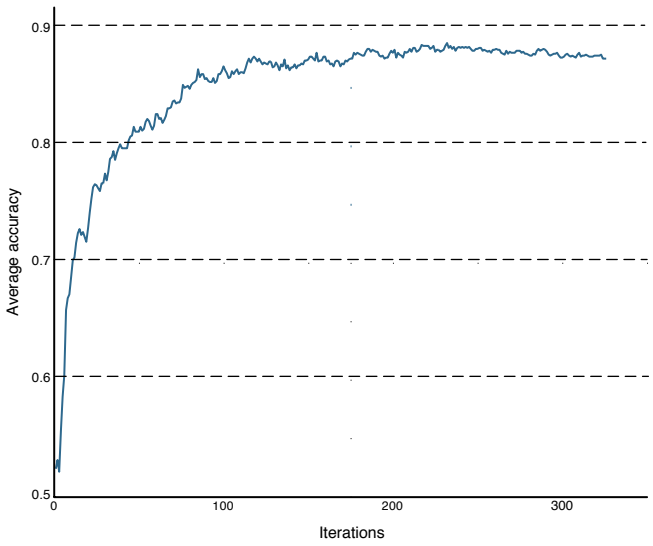


Figura 4.9: Precisión obtenida mediante aprendizaje activo durante las iteraciones, utilizando voz de hombre y de mujer del conjunto WaSeP. Cada iteración representa 10 nuevas etiquetas utilizadas para entrenamiento.

tenido es exactamente igual al obtenido por humanos. En el caso de EmoDB, un 77 % obtenido mediante reconocimiento automático se muestra ligeramente inferior al 84.7 % obtenido por humanos. Los valores medios para ambos conjuntos de datos son muy similares. Sin embargo, existen ciertos patrones que parecen diverger bastante. Primeramente, con respecto a WaSeP (comparar Cuadros 2.1 y 4.3) muchos de los errores cometidos por humanos son debidos a la elección de la clase neutral, lo que induce a pensar que los humanos tienen tendencia a votar por una clase que no presenta clara evidencia de la emoción. La máquina no elige neutro con tanta frecuencia, en parte debido al hecho de que la palabra neutro no presenta ningún significado para ella.

También es observado que felicidad es reconocida con dificultad en los experimentos conducidos con ambos conjuntos de datos. En el caso de la percepción humana, esto es distinto. No obstante, existe evidencia en la literatura de que la felicidad es a menudo difícil de reconocer, tal y como se concluye en Scherer et al. (2001). En este caso, únicamente un 48 % pudo ser alcanzado. En Wendt (2007) también se observa que la precisión de reconocimiento para humanos con respecto a las grabaciones de felicidad en el caso de hombres es del 66 %, mostrando una mayor confusión hacia neutro. Este hecho viene confirmado en el Cuadro 4.3.

Enojo es sendos tests de percepción humana la emoción mejor reconocida, mientras que la clasificación automática no alcanza esos niveles de corrección. Por otro lado, el clasificador alcanza grandes precisiones para expresiones de tristeza, las cuáles lo humanos suelen confundir frecuentemente para ambos conjuntos de datos. Una vez más, en Wendt (2007) se reporta una diferencia entre géneros: las expresiones de tristeza en mujeres sólo son reconocidas al 65 % y principalmente confundidas con neutral.

Con la precisión más baja en el caso de WaSeP encontramos repugnancia, lo que está de acuerdo con Van Bezooyen (1984); Scherer et al. (1991) También está entre las precisiones más bajas en el caso de EmoDB, superando únicamente a felicidad. Este efecto era muy presente en nuestros experimentos con los grupos de rasgos estándar. El diseño de un nuevo conjunto de rasgos, tal y como se explica en la Sección 3.1.7, que proporciona buenos resultados para el caso de repugnancia, permite alcanzar una mejora en la fusión en torno a un 20 %.

También debe ser observado en las Figuras 4.2 y 4.3 que la precisión obtenida con la fusión de todos los rasgos supera a cada uno de ellos individualmente. Esto prueba que la solución propuesta para realizar la fusión produce buenos resultados a pesar de ser una aproximación muy sencilla.

En referencia a los experimentos de entrenamiento semi-supervisado mediante k-NN, un análisis de los resultados obtenidos (ver Figura 4.6) muestra que el uso de muestras sin etiqueta en el proceso de entrenamiento no mejora la baseline. En este caso, la baseline fue reducida para simular una situación con pequeñas cantidad de datos disponibles (unas 20 muestras por categoría). Para esta referencia, una precisión del 73 % es obtenida, considerando independencia

de género. Como ya se ha comentado en la Sección 4.2.1, un barrido sobre el parámetro p fue conducido, encontrando su máximo en $p = 0{,}8$. Éste, sin embargo, no representa una mejora con respecto a la referencia. Dada la gran cantidad de muestras automáticamente etiquetadas para el entrenamiento, cabe pensar que el estilo en que los experimentos han sido diseñados no es óptimo. Puede existir distintos motivos para esto, como una mala elección de la medida de confianza o una cantidad excesiva de error presente en las etiquetas automáticamente generadas. Suponiendo que la medida de confianza utilizada es válida, mejores resultados han de ser esperados si las etiquetas artificiales se generan con mayor precisión. Para probar esta suposición, un segundo experimento ha sido realizado donde se pretendía reducir el error introducido artificialmente en las etiquetas. Después, el conjunto de etiquetas de referencia utilizadas para entrenar los SVMs es reducido, de modo que la mayor parte de las etiquetas de entrenamiento son artificiales. Tal y como se observa en la Figura 4.8, si sólo una pequeña cantidad de error es introducida artificialmente, cierta mejora en la precisión puede ser alcanzada. Es de suponer que con un algoritmo de etiquetado que inserte menor cantidad de error que k-NN, puede ser posible usar entrenamiento semi-supervisado con buenos resultados de precisión. Con etiquetas artificiales que contienen poco error, la medida de confianza ha demostrado funcionar correctamente en nuestros experimentos.

En oposición a los resultados poco esperanzadores obtenidos con el entrenamiento semi-supervisado, el entrenamiento activo ha demostrado ser una muy buena solución para reducir la cantidad de datos etiquetados necesarios. Se puede observar que tras unas 60 iteraciones (unas 100 muestras por cada clase), la precisión alcanza un nivel similar al alcanzado con el aprendizaje supervisado usando el doble de etiquetas para entrenar. Esto implica una gran reducción de la cantidad de etiquetado necesario, demostrando que esta solución funciona y produce buenos resultados. En la Figura 4.9 se observa que tras cierta iteración (en torno a la número 100), la adición de nuevas muestras no conlleva una mejora en la precisión. Es posible afirmar entonces que el aprendizaje activo funciona bien y puede reducir significativamente la cantidad de etiquetas necesarias sin penalizar los resultados obtenidos.

5 Resumen y Conclusiones

5.1. Sumario

Este estudio presenta un nuevo estilo para solucionar el problema de la clasificación automática de emociones basada en voz. Una combinación de grupos de rasgos estándar han sido descritos y un nuevo conjunto ha sido desarrollado para representar mejor la percepción humana. Un proceso de alineamiento de rasgos basado en la verosimilitud de HMMs también ha sido incluído en el estudio, representando una novedad para el caso de clasificación de emociones. Con esta técnica, secuencias de rasgos de diferentes longitudes pueden codificarse en vectores de dimensiones fijas, permitiendo la comparación entre ellas y también con rasgos estáticos. Clasificadores SVM multi-clase fueron entrenados en modo supervisado y utilizados para evaluar la precisión del sistema. Para ello, una técnica de fusión ha sido propuesta, proporcionando buenos resultados. Comparación de los resultados obtenidos con los test de percepción humanos también ha sido conducida, mostrando que buenas precisiones pueden ser obtenidas, mostrando confusiones entre clases similares para ambos casos. Este hecho prueba que la combinación de rasgos propuesta es suficientemente capaz de representar bien las capacidades de percepción humanas.

También hay contribuciones al campo del aprendizaje parcialmente supervisado, donde el aprendizaje semi-supervisado y activo han sido estudiados dentro del esquema de clasificación propuesto. Una medida de confianza ha sido propuesta para ambos casos, representando el nivel de corrección en el primer caso, y la distancia a las fronteras de decisión en el otro. Para los experimentos de aprendizaje semi-supervisado, es interesante conocer la cantidad de error que ha sido introducido, teniendo mucha relevancia las muestras con una confianza alta. Por el contrario, en el caso de aprendizaje activo, la medida de confianza es utilizada para encontrar las muestras más informativas, representadas por un valor de confianza bajo. La definición de la medida de confianza ha sido un factor importante puesto que este tipo de soluciones requieren una medida de este tipo para poder decidir en cada instante qué hacer con cada una de las muestras disponibles. En los experimentos semi-supervisados, los resultados han demostrado que para el caso de sets de referencia muy pequeños, k-NN introduce demasiado error, provocando que el sistema vea reducida su precisión.

No obstante, un segundo experimento en el que se redujo la cantidad de error artificial mostró que la precisión puede ser aumentada con etiquetas artificiales. Con este fin, una técnica mejor que k-NN debe ser encontrada para el caso en el que pocas muestras de referencia estén disponibles. Como contraste, el aprendizaje activo ha alcanzado muy buenos resultados y se muestra como una muy buena opción para reducir los costes derivados del etiquetado, reduciendo la cantidad de muestras necesarias en los pasos de entrenamiento. La medida de confianza propuesta asegura un incremento continuo en la precisión del sistema, comenzando con un conjunto muy reducido de muestras etiquetadas y solicitando al experto nuevas etiquetas para las muestras con menos confianza en cada iteración.

5.2. Cuestiones Abiertas y Trabajo Futuro

Tal y como se explicó en el Capítulo 2, los datos utilizados para este trabajo provienen de conjuntos estándar utilizados ampliamente por la comunidad investigadora. Sin embargo, ambos presentan el inconveniente de ser únicamente emociones actuadas, lo cuál añade una componente no realista a nuestros experimentos. La creación de un conjunto de datos grabado en una situación realmente emocional sería deseable para poder representar con mayor precisión el comportamiento afectivo.

Durante el estudio de los distintos grupos de rasgos utilizados normalmente para el reconocimiento de voz así como para otras tareas basadas en este tipo de datos, se observo que prácticamente todos ellos están basados en la energía de la señal. Algunos se basan en la distribución de la energía sobre las distintas ventanas y otras en la distribución de la energía en una misma ventana. Sin embargo, esto implica que una gran cantidad de información está siendo perdida puesto que la representación de una señal en el dominio de Fourier no sólo contiene energía, sino también información de fase. Algunas pruebas han sido realizadas durante este estudio donde las características de amplitud o de fase han sido eliminadas de la señal, obteniendo señales con información únicamente de una de las dos. Los resultados mostraron que la reconstrucción obtenida únicamente a partir de las propiedades de la energía es incomprensible, mientras que la reconstrucción basada en la información de la fase aún puede ser entendida (Hayes et al., 1980). Por este motivo, podría ser interesante en trabajos futuros el desarrollo de un conjunto de rasgos que combine propiedades de fase con energía, de modo que puedan representar mejor las características de la señal.

Con respecto a la fusión utilizada para combinar los distintos conjuntos de rasgos, una simple multiplicación y normalización de las distintas salidas de los clasificadores ha sido considerada, dando muy buenos resultados. No obstante, podrían considerarse implementaciones más complejas que pudieran mejorar los resultados tendiendo en cuenta las características de cada conjunto de rasgos individualmente. Por ejemplo, los MFCCs producen en general muy buenos

resultados (ver Figuras 4.2 y 4.3) y otros como la Energía, no tan buenos. Por tanto, estas conclusiones podrían ser usadas para definir distintas combinaciones de pesos que ponderaran la salida de cada clasificador.

La medida de confianza propuesta en la Sección 3.6.1 ha mostrado dar buenos resultados siendo su cómputo extremadamente sencillo. Aún así, medidas más robustas podrían tal vez ser implementadas considerando los resultados mostrados en 4.5. Puesto que las etiquetas que nuestro sistema utiliza son difusas, tendría sentido considerar no solamente la clase con un mayor grado de pertenencia, sino también la combinación con otras clases.

En cuanto a los entrenamientos basados en una técnica semi-supervisada, se ha podido concluir que kNN por sí solo no es capaz de producir etiquetas con suficiente confianza cuando el set de referencia es muy reducido. Para solucionar este problema, podría ser posible utilizar técnicas de entrenamiento cooperativo, donde una cooperación iterativa entre el algoritmo kNN y las salidas de los SVMs podría ser definida y utilizada para mejorar las etiquetas automáticamente generadas. Igualmente, también podría ser posible encontrar diferentes algoritmos que produzcan mejores resultados, tales como Naive Bayes (NB), Fuzzy Assignmemt Procedure for Nominal Sorting (PROAFTN por sus siglas en Francés) o incluso una etapa extra de SVMs entrenada con el set de referencia.

Índice de cuadros

Índice de figuras

Bibliografía

Banse, R., Scherer, K. R., 1996. Acoustic profiles in vocal emotion expression. Journal of Personality and Social Psychology 70 (3), 614–636.

Bicego, M., Murino, V., Figueiredo, M., 2003. Similarity-based clustering of sequences using hidden markov models. In: Perner, P., Rosenfeld, A. (Eds.), Machine Learning and Data Mining in Pattern Recognition. Vol. 2734. Springer, pp. 95–104.

Bishop, C. M., October 2006. Pattern Recognition and Machine Learning (Information Science and Statistics), 1st Edition. Springer.

Blum, A., Mitchell, T., 1998. Combining labeled and unlabeled data with co-training. In: Proceedings of the eleventh annual conference on Computational learning theory. COLT' 98. ACM, New York, NY, USA, pp. 92–100.

Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W., Weiss, B., 2005. A database of german emotional speech. In: Proceedings of Interspeech 2005. ISCA, pp. 1517–1520.

Cedergren, H. J., Perreault, H., 1994. Speech rate and syllable timing in spontaneous speech. In: Third International Conference on Spoken Language Processing (ICSLP 94). IEEE, pp. 1087–1090.

Crystal, T. H., House, A. S., 1990. Articulation rate and the duration of syllables and stress groups in connected speech. Journal of The Acoustical Society of America 88, 101–112.

Druck, G., Mann, G., McCallum, A., 2008. Learning from labeled features using generalized expectation criteria. In: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '08. ACM, New York, NY, USA, pp. 595–602.

Duda, R. O., Hart, P. E., Stork, D. G., 2001. Pattern Classification, 2nd Edition. Wiley, New York.

Fang, Z., Guoliang, Z., Zhanjiang, S., 2001. Comparison of different implementations of mfcc. J. Comput. Sci. Technol. 16 (6), 582–589.

Gobl, C., Bennett, E., Chasaide, A. N., Sept. 2002. Expressive synthesis: how crucial is voice quality? In: IEEE Workshop on Speech Synthesis, 2002. IEEE, pp. 91–94.

Hayes, M., Lim, J., Oppenheim, A., dec 1980. Signal reconstruction from phase or magnitude. Acoustics, Speech and Signal Processing, IEEE Transactions on 28 (6), 672 – 680.

Hermansky, H., Apr. 1990. Perceptual linear predictive (PLP) analysis of speech. Acoustical Society of America Journal 87, 1738–1752.

Hermansky, H., Morgan, N., 1994. Rasta processing of speech. IEEE Transactions on Speech and Audio Processing, special issue on Robust Speech Recognition 2, 578–589.

Kahsay, L., Schwenker, F., Palm, G., 2005. Comparison of multiclass svm decomposition schemes for visual object recognition. In: Kropatsch, W. G., Sablatnig, R., Hanbury, A. (Eds.), Pattern Recognition. Vol. 3663 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp. 334–341.

Kane, J., Kane, M., Gobl, C., 2010. A spectral lf model based approach to voice source parameterisation. In: Proceedings of Interspeech 2010. ISCA, pp. 2606–2609.

Kane, J., Scherer, S., Gobl, C., under review. Glottal source parameterisation in the frequency domain based on the lf model spectrum. Speech Communication: Special Issue on Advanced Voice Function Assessment.

Keltner, D., Ekman, P., 2003. Handbook of Affective Sciences - Introduction: Expression of Emotion. Affective Science. Oxford University Press, Ch. 21, pp. 411–414.

Keltner, D., Ekman, P., Gonzaga, G. C., Beer, J., 2003. Handbook of Affective Sciences - Facial expression of emotion. Affective Science. Oxford University Press, Ch. 22, pp. 415–432.

Kuncheva, L. I., 2001. Using measures of similarity and inclusion for multiple classifier fusion by decision templates. Fuzzy Sets and Systems 122 (3), 401–407.

Kuncheva, L. I., 2004. Combining pattern classifiers: methods and algorithms. Wiley.

Kuncheva, L. I., Bezdek, J. C., Duin, R. P. W., 2001. Decision templates for multiple classifier fusion: an experimental comparison. Pattern Recognition 34 (2), 299–314.

Li, D., Sethi, I. K., Dimitrova, N., McGee, T., 2001. Classification of general audio data for content-based retrieval. Pattern Recognition Letters 22 (5), 533 – 544.

Logan, B., 2000. Mel frequency cepstral coefficients for music modeling. In: In International Symposium on Music Information Retrieval.

Lomasky, R., Brodley, C. E., Aernecke, M., Walt, D., Friedl, M. A., 2007. Active class selection. In: ECML'07. pp. 640–647.

Mayergoyz, I., 2003. Mathematical Models of Hysteresis and their Applications. Springer.

Pearson, K., 1901. On lines and planes of closest fit to systems of points in space, 559572.

Pfitzinger, H., Burger, S., Heid, S., 1996. Syllable detection in read and spontaneous speech. In: Proceedings of The 4th International Conference on Spoken Language Processing (ICSLP'96). Vol. 2. Philadelphia, pp. 1261–1264.

Scherer, K. R., Banse, R., Wallbott, H. G., 2001. Emotion inferences from vocal expression correlate across languages and cultures. Journal of Cross-Cultural Psychology 32, 76–92.

Scherer, K. R., Banse, R., Wallbott, H. G., Goldbeck, T., 1991. Vocal cues in emotion encoding and decoding. Motivation and Emotion 15, 123–148.

Scherer, K. R., Johnstone, T., Klasmeyer, G., 2003. Handbook of Affective Sciences - Vocal expression of emotion. Affective Science. Oxford University Press, Ch. 23, pp. 433–456.

Scherer, S., Kane, J., Gobl, C., Schwenker, F., under review. Investigating fuzzy-input fuzzy-output support vector machines for robust voice quality classification. IEEE Transactions on Audio, Speech and Language Processing.

Scherer, S., Oubbati, M., Schwenker, F., Palm, G., 2008. Real-time emotion recognition from speech using echo state networks. In: Proceedings of the 3rd IAPR workshop on Artificial Neural Networks in Pattern Recognition (ANNPR'08). Springer, Berlin, Heidelberg, pp. 205–216.

Scherer, S., Schwenker, F., Palm, G., Sept. 2007. Classifier fusion for emotion recognition from speech. In: 3rd IET International Conference on Intelligent Environments 2007 (IE07). IEEE, pp. 152–155.

Settles, B., 2009. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison.

Thiel, C., 2009. Multiple classifier systems incorporating uncertainty. Ph.D. thesis, Ulm University.

Thiel, C., Giacco, F., Schwenker, F., Palm, G., 2009. Comparison of neural classification algorithms applied to land cover mapping. In: Proceeding of the 2009 conference on New Directions in Neural Networks. IOS Press, Amsterdam, The Netherlands, The Netherlands, pp. 254–263.

Thiel, C., Scherer, S., Schwenker, F., 2007. Fuzzy-input fuzzy-output one-against-all support vector machines. In: 11th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems (KES 2007). Vol. 3 of Lecture Notes in Artificial Intelligence. Springer, pp. 156–165.

Van Bezooyen, R., 1984. Characteristics and Recognizability of Vocal Expressions of Emotion. Foris Pubns USA.

Vlasenko, B., Schuller, B., Wendemuth, A., Rigoll, G., 2007. Frame vs. turn-level: Emotion recognition from speech considering static and dynamic processing. In: Proceedings of the 2nd international conference on Affective Computing and Intelligent Interaction (ACII'07). Springer-Verlag, Berlin, Heidelberg, pp. 139–147.

Wagner, J., Vogt, T., André, E., 2007. A systematic comparison of different hmm designs for emotion recognition from acted and spontaneous speech. In: Proceedings of the 2nd international conference on Affective Computing and Intelligent Interaction (ACII'07). Springer-Verlag, Berlin, Heidelberg, pp. 114–125.

Wendt, B., 2007. Analysen Emotionaler Prosodie. Vol. 20 of Hallesche Schriften zur Sprechwissenschaft und Phonetik. Peter Lang Internationaler Verlag der Wissenschaften.

Wendt, B., Scheich, H., 2002. The "Magdeburger Prosodie Korpus a spoken language corpus for fMRI-studies. In: Speech Prosody 2002. SProSIG, pp. 699–701.

Yanushevskaya, I., Gobl, C., Ní Chasaide, A., 2008. Voice quality and loudness in affect perception. In: Speech Prosody 2008. Campinas, Brazil.

Zhu, X., 2005. Semi-supervised learning literature survey. Tech. Rep. 1530, Computer Sciences, University of Wisconsin-Madison.