Aplicación de la estimación no paramétrica a la fusión de rankings

Alejandro Bellogín Kouki
alejandro.bellogin@uam.es

4 de febrero de 2008

1. Introducción

La fusión de rankings es una técnica aplicada en múltiples tareas de Recuperación de Información, por ello se hace necesario realizarla con la mayor fiabilidad posible. Este trabajo es una propuesta para mejorar la fase de normalización, siguiendo el trabajo de Fernández et al. en [5] donde se extraen las distribuciones de cada una de las fuentes implicadas para combinarlas, en lugar de combinar directamente las puntuaciones; nosotros proponemos una mejora a la hora de estimar dichas distribuciones: emplear técnicas no parámetricas. Al utilizar estas técnicas esperamos que se mejoren los resultados, aunque somos conscientes de que también añade algunos problemas, como la estimación del ancho de banda y el factor de suavizado óptimos.

2. Estimación no parámetrica y suavizado

Dada una variable aleatoria (X), el conocimiento de su **función de densidad** nos ayuda para descubrir muchas características de la variable, como por ejemplo cuántas observaciones caen en un determinado conjunto de X:

$$P(a < X < B) = \int_a^b f(x)dx$$

No obstante, en la práctica no solemos conocer la función de densidad (f) de X, sino un conjunto n de observaciones $\{X_i\}_{i=1}^n$ de las cuales podemos asumir que son observaciones independientes idénticamente distribuidas de una variable aleatoria con función de distribución desconocida f. Por lo tanto, una tarea importante es encontrar la función de densidad asociada a dicha función de distribución, para lo cual hay dos alternativas fundamentales:

- Aproximación paramétrica
- Aproximación no parámetrica

En el primer caso, dotamos a la función de densidad de un conjunto finito de parámetros, de manera que estimar la densidad es equivalente a estimar los parámetros, para ello debemos asumir que la función de densidad es similar

a alguna familia de funciones de densidad (por ejemplo, a una normal o una Poisson). En el segundo caso debemos estimar la curva por completo, lo cual es necesario cuando no tenemos información precisa sobre la forma o el tipo de la densidad auténtica o, aún cuando se conoce, se quiere obtener otra perspectiva.

En nuestro caso estamos interesados en la segunda opción, por lo que presentaremos distintas técnicas para lograr nuestro objetivo: obtener la función de densidad sin la utilización de ningún parámetro, sólo a partir de los datos iniciales de las observaciones.

2.1. Técnicas

A continuación presentamos distintas técnicas no parámetricas para estimar la función de densidad [7, 3]. El orden presentado se debe tanto al orden histórico como a la complejidad que implican, y, por ello mismo, al campo de aplicación que ha tenido cada uno, en virtud de esto último, ya que el histograma es la técnica más sencilla de aplicar, se utiliza mucho y surgió antes que el resto de técnicas, es la primera que vamos a revisar.

2.1.1. Histograma

Para calcular un histograma que sirva para estimar una función de densidad unidimensional, hay que realizar los siguientes pasos:

1. Dividir la recta real en clases o **subintervalos** (*bins*)

$$B_j = [x_0 + (j-1)h, x_0 + jh), j \in \mathbb{Z}$$

   donde $h > 0$ es la longitud de cada clase o anchura del compartimento, y x_0 es el origen del histograma

2. Contar cuántos datos caen en cada clase

Esto es equivalente a:

$$\hat{f}_h(x) = (nh)^{-1} \sum_{i=1}^{n} \sum_{j} I(X_i \in B_j) I(x \in B_j)$$

donde $I(M)$ es la función indicatriz en M, es decir, vale 1 en M. Para ver la equivalencia basta pensar que $\sum_i I(X_i \in B_j)$ cuenta los datos que aparecen en un determinado compartimento (por ejemplo, si tenemos para unos determinados datos que $X_1 = 26.4$, $X_2 = 82.4$, $X_3 = 78.1$ y $\{B_j\} = \{[0, 50), [50, 100)\}$, entonces $\sum_i I(X_i \in B_1) = 1, \sum_i I(X_i \in B_2) = 2$).

Es evidente que un parámetro muy importante en este proceso es el ancho h de cada subintervalo, ya que variándolo encontramos diferentes formas de $\hat{f}_h(x)$. De hecho, se puede ver que interesa que $h \to 0$ para mantener el sesgo (*bias*) bajo, lo cual es incompatible con asegurar que suficientes observaciones caigan dentro de cada clase: $nh \to \infty$. Estas observaciones se desprenden de la fórmula del error cuadrático medio (interesa que $MSE(f_h(x)) \to 0$ para concluir que $\hat{f}_h(x)$ es un estimador consistente de $f(x)$):

$$MSE(\hat{f}_h(x)) = \frac{1}{nh} f(x) + \left(\left(j - \frac{1}{2} \right) h - x \right)^2 f' \left(\left(j - \frac{1}{2} \right) h \right)^2 + o(h) + o\left(\frac{1}{nh} \right)$$

Se puede ver que la fórmula anterior es díficil de calcular en la práctica ya que necesita la función f, desconocida, que tratamos de estimar, por ello, se define el error cuadrático medio integrado:

$$MISE(\hat{f}_h) = \int_{-\infty}^{+\infty} MSE(f_h\hat{(}x))dx = (nh)^{-1} + \frac{h^2}{12}||f'||_2^2 + o(h^2) + o((nh)^{-1})$$

El problema encontrado ahora es que necesitaríamos conocer f', no obstante, se puede utilizar el *método plug-in*, que consiste en estimar $||f'||_2^2$ e introducirlo en la última fórmula escrita. Desafortunadamente, en esta estimación también aparecen problemas con el tamaño del ancho, por lo que lo que se suele hacer normalmente es tomar una distribución de referencia[1] para $||f'||_2^2$, y de esta manera calcular el ancho óptimo h_0 que minimiza $MISE(\hat{f}_h)$, el cual en teoría es $h_0 \sim n^{-1/3}$

2.1.2. WARPing

Un problema con respecto a los histogramas que no hemos mencionado es la fuerte dependencia en la elección de x_0 (origen). Una solución a este inconveniente es calcular la media de los histogramas realizados con distinto origen, de manera que el histograma medio no depende de x_0, es por ello que se le llama a esto histograma medio desplazado (ASH o *Average Shifted Histogram*). La generalización de este método se conoce como WARPing (*Weighted Averaging of Rounded Points*), en él escogemos subintervalos más pequeños, y en vez de una sola dimensión, dos:

$$B_{j,l} = \left[\left(j - 1 + \frac{1}{M}\right)h, \left(j + \frac{l}{M}\right)h\right), l \in \{0, \cdots, M-1\}$$

Básicamente lo que se hace es mover cada B_j una cantidad de $\frac{lh}{M}$ a la derecha, por lo que tenemos ahora M histogramas basados en estos subintervalos:

$$\hat{f}_{h,l}(x) = (nh)^{-1} \sum_{i=1}^{n} \left(\sum_j I(x \in B_{j,l}) I(X_i \in B_{j,l}) \right)$$

La idea de WARPing es calcular la media sobre estos histogramas:

$$\begin{aligned} \hat{f}_h(x) &= M^{-1} \sum_{l=0}^{M-1} (nh)^{-1} \sum_{i=1}^{n} \left(\sum_j I(x \in B_{j,l}) I(X_i \in B_{j,l}) \right) \\ &= n^{-1} \sum_{i=1}^{n} \left((Mh)^{-1} \sum_{l=0}^{M-1} \sum_j I(x \in B_{j,l}) I(X_i \in B_{j,l}) \right) \end{aligned}$$

Si llamamos $B_z^* = \left[\frac{z}{M}h, \frac{z+1}{M}h\right)$ obtenemos la siguiente expresión:

$$\hat{f}_h(x) = (nh)^{-1} \sum_j I_{B_j^*} \sum_{k=1-M}^{M-1} w_M(k) n_{j+k}$$

[1] En particular, una distribución de referencia comúnmente utilizada es la distribución normal para la cual

$$f'(u) = N'(u) = -uN(u), \quad N(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}$$

donde $n_k = \sum_{i=1}^{n} I_{B_k^*}(X_i), w_M(k) = 1 - \frac{|k|}{M}$. De hecho, el haber separado la expresión de w_M en la fórmula nos lleva a una clase mayor de estimadores: los estimadores de densidad kernel.

2.1.3. Estimadores Kernel

Este tipo de estimadores se deben a Rosenblatt (1956); en la literatura se pueden encontrar también con el nombre de **ventanas de Parzen** (*Parzen windows*) y su característica principal es que cada punto observado se pesa mediante una función kernel o ventana (dependiendo del nombre con el que se llame a estos estimadores, en adelante nos referiremos a ellos como kernel). Estas funciones kernel tienen un parámetro que regula el grado de suavidad del estimador: el **ancho de banda** (*bandwidth*) denotado por h, en concreto, dada la función kernel genérica K se tiene la función kernel K_h:

$$K_h(x) = \frac{1}{h} K\left(\frac{x}{h}\right)$$

Si ahora tomamos la media de estas funciones en las observaciones obtenemos el estimador de densidad kernel[2]:

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - X_i) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right)$$

Hay que decir que una función kernel debe ser:

- No negativa, con valores en los números reales e integrable.
- Simétrica con respecto al cero: $K(-u) = K(u)\ \forall u$
- Función de densidad, es decir: $\int_{-\infty}^{+\infty} K(u)du = 1$.

En particular, estas propiedades inducen otras respectivas en el estimador kernel que acabamos de definir:

- Es una densidad.
- Hereda la propiedad de suavidad, es decir, si K es n veces diferenciable con continuidad, $\hat{f}_h(x)$ también es diferenciable n veces con continuidad.
- No depende de la elección del origen (al contrario que los histogramas). En particular, dado un ancho de banda h y una función kernel K, la función de densidad kernel correspondiente es única para un conjunto de datos.

[2]Las ventanas de Parzen vienen motivadas por la función ventana φ, que define un hipercubo unidad d-dimensional centrado en el origen (con longitud de cada arista h), de manera que el número de muestras en el hipercubo viene dado por $\sum_{i=1}^{n} \varphi\left(\frac{x-X_i}{h}\right)$, el estimador es

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{V}\varphi\left(\frac{x - X_i}{h}\right), V = h^d$$

Algunas funciones kernel comúnmente utilizadas[3]:

Kernel	$K(u)$
Uniform	$\frac{1}{2}I(\|u\| \leq 1)$
Triangle	$(1-\|u\|)I(\|u\| \leq 1)$
Epanechnikov	$\frac{3}{4}(1-u^2)I(\|u\| \leq 1)$
Quartic	$\frac{15}{16}(1-u^2)^2I(\|u\| \leq 1)$
Triweight	$\frac{35}{32}(1-u^2)^3I(\|u\| \leq 1)$
Gaussian	$\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}u^2\right)$
Cosinus	$\frac{\pi}{2}\cos\left(\frac{\pi}{2}u\right)I(\|u\| \leq 1)$

3. Fusión de rankings

La fusión o agregación de rankings se hace necesaria cuando se quieren combinar varias listas de resultados de distintas fuentes en una sola, desconociendo el proceso que ha seguido cada una de las fuentes para producir dichas listas, los datos que ha utilizado, el rango de puntuaciones para cada elemento, etc. De esta manera, tiene aplicación en sistemas que utilicen varios criterios de búsqueda para los documentos, en metabuscadores, sistemas que realizan búsqueda personalizada (hay que combinar los resultados provenientes sólo de la personalización con los de la búsqueda basada en la consulta), ...

Existen algunos problemas inherentes a la fusión de rankings: cada fuente ha podido utilizar un método distinto para devolver los documentos de una determinada manera (por semejanza o por diferencia con respecto a la consulta, evaluando las frecuencias de los términos o los modelos probabilísticos subyacentes), además, el rango en el que se mueven las puntuaciones para cada documento (en el supuesto en el que se conozcan) no tiene por qué ser el mismo para todas las fuentes, por lo que las puntuaciones no son equivalentes y complica el proceso de agregación de rankings. Por estas razones el proceso de fusión de rankings se divide en normalización (transformación de los datos en un dominio común para poder realizar la siguiente fase) y combinación (método por el que se juntan las distintas listas normalizadas en una sola), además existen técnicas en cada una de estas fases que utilizan la posición (ranking) del documento o la puntuación devuelta por la fuente para cada documento, también se distinguen unas de otras en la necesidad o no de datos de entrenamiento.

[3]Hay que notar que existe una relación entre los pares del mismo orden $(K_i, h_i) \mapsto (K_j, h_j)$ de manera que $A-MISE(K_j, h_j) = A_MISE(K_i, h_i) * C$, con C factor que depende de las funciones kernel involucradas, si $h_j = h_i \frac{\delta_j^*}{\delta_i^*}$ donde δ_j^* es el *ancho de banda canónico* del kernel K_j. Es decir, dado un kernel y un ancho de banda, se puede encontrar el ancho de banda tal que con otro kernel (de su mismo orden) se obtenga el mismo grado de suavizado (en [8] se dan las cantidades δ_j^*/δ_i^*)

En lo sucesivo usaremos la siguiente notación:

$\Omega = \{d_1, \cdots, d_n\}$	Universo de documentos
$\mathcal{R} = \{\tau_1, \cdots, \tau_k\}$	Rankings a combinar)
$\tau(d)$	Posición del documento d en el ranking τ
$s_\tau(d)$	Puntuación de d en el ranking τ
$\Omega\tau \subset \Omega$	Documentos devueltos por τ
$\Omega_\mathcal{R} = \cup_{\tau\in\mathcal{R}}\Omega_\tau \subset \Omega$	Documentos devueltos por algún τ en $\mathcal{R}$
$\bar{s}_\tau(d)$	Puntuación normalizada de d en τ
$s_\mathcal{R}(d)$	Puntuación combinada de todos los rankings de $\mathcal{R}$
σ^2	Varianza

3.1. Métodos de normalización

Este proceso es muy importante ya que mueve los datos iniciales a un dominio común, donde se apliquen las restantes fases.

Estos métodos se pueden dividir según se apliquen a posiciones en el ranking o a puntuaciones (devueltas por la fuente). Además, entre los métodos que veremos hay algunos que necesitan datos de entrenamiento, aunque para la gran mayoría no es así.

Basados en posición (*rank based*):

$$\text{Rank-sim} \quad : \quad \bar{s}_\tau(d) = 1 - \frac{\tau(d)-1}{|\Omega_\tau|}$$

$$\text{Borda} \quad : \quad \bar{s}_\tau(d) = \begin{cases} 1 - \frac{\tau(d)-1}{|\Omega|} & , \quad d \in \Omega \\ \frac{|\Omega|-|\Omega_\tau|+1}{2|\Omega|} & , \quad d \notin \Omega \end{cases}$$

$$\text{Bayes} \quad : \quad \bar{s}_\tau(d) = \log \frac{P(\tau(d)|d \text{ es relevante})}{P(\tau(d)|d \text{ no es relevante})}$$

El único de estos métodos que precisa datos de entrenamiento es Bayes, el cual necesita juicios de relevancia *a priori* (como pueden ser los que se incluyen en las colecciones TREC).

Basados en puntuación (*score based*):

$$\text{Standard} \quad : \quad \bar{s}_\tau(d) = \begin{cases} \frac{s_\tau(d) - \text{mín}_{d'\in\Omega_\tau} s_\tau(d')}{\text{máx}_{d'\in\Omega_\tau} s_\tau(d') - \text{mín}_{d'\in\Omega_\tau} s_\tau(d')} & , \quad d \in \Omega \\ 0 & , \quad d \notin \Omega \end{cases}$$

$$\text{Sum} \quad : \quad \bar{s}_\tau(d) = \begin{cases} \frac{s_\tau(d) - \text{mín}_{d'\in\Omega_\tau} s_\tau(d')}{\sum_{d'\in\Omega_\tau} s_\tau(d') - \text{mín}_{d'\in\Omega_\tau} s_\tau(d')} & , \quad d \in \Omega \\ 0 & , \quad d \notin \Omega \end{cases}$$

$$\text{ZMUV} \quad : \quad \bar{s}_\tau(d) = \begin{cases} \frac{s_\tau(d)-\mu}{\sigma^2} & , \quad d \in \Omega, \sigma^2 \neq 0 \\ 0 & , \quad d \in \Omega, \sigma^2 = 0 \\ -2 & , \quad d \notin \Omega \end{cases}$$

$$\text{2MUV} \quad : \quad \bar{s}_\tau(d) = \begin{cases} 2 - \frac{s_\tau(d)-\mu}{\sigma^2} & , \quad d \in \Omega, \sigma^2 \neq 0 \\ 0 & , \quad d \in \Omega, \sigma^2 = 0 \\ 0 & , \quad d \notin \Omega \end{cases}$$

$$\text{Manmatha} \quad : \quad \bar{s}_\tau(d) = P(y \text{ es relevante } | s_\tau(y) = s_\tau(x))$$

Ninguno de los métodos basados en puntuación que acabamos de enumerar necesita datos de entrenamiento. No obstante, el de Manmatha [11] asume que los conjuntos de documentos no relevantes siguen una distribución exponencial, mientras que los de relevantes siguen una gaussiana; además, hace uso del algoritmo EM (*Expectation-Maximization*) para aproximar los parámetros necesarios de las distribuciones recién mencionadas. Por su parte, los otros métodos fueron propuestos por Montague y Aslam en [12] y se diferencian en el parámetro que desplazan a cero (unas veces el mínimo, otras la media) y en el valor que le dan a los documentos no recuperados; esto provoca que los dos métodos que no son sensibles a desviaciones externas son ZMUV y 2MUV.

3.1.1. Normalización probabilística basada en puntuación

En [5] encontramos un método novedoso de normalización basada en puntuación, el cual intenta evitar las distorsiones provocadas cuando se combinan distintos rankings con sesgos diferentes. Para ello, tienen en cuenta la distribución de las puntuaciones, asociando posteriormente estas distribuciones a una Distribución Óptima de Puntuaciones (OSD, del inglés *optimal score distribution*), de manera que las puntuaciones sean comparables y se puedan combinar sin sufrir ruido debido al sesgo.

El modelo propuesto asume una función de puntuaciones ideal no sesgada $r(x)$ con valores en el intervalo $[0, 1]$ y distribución acumulada $\bar{F}$. Por otro lado, dada una función de puntuación $s_\tau(x)$ y su distribución acumulada F_τ, entonces la función normalizada es $\bar{s}_\tau = \bar{F}^{-1} \circ F_\tau \circ s_\tau$, la cual es solución de $P(s_\tau(y) \leq \bar{s}_\tau(x)) = P(r(y) \leq \bar{s}_\tau(x))$. Esto se consigue por medio de los siguientes pasos:

1. Para cada fuente o ranking τ se calcula la distribución acumulada F_τ de los valores s_τ.

2. Se construye una OSD estrictamente creciente $\bar{F} : [0, 1] \to [0, 1]$.

3. Para cada $x \in \Omega$ y $\tau \in \mathcal{R}$ se asocia la puntuación de cada fuente a la OSD (normalización):

$$s_\tau(x) \longrightarrow \bar{s}_\tau(x) = \bar{F}^{-1} \circ F_\tau \circ s_\tau(x)$$

4. Unión de las puntuaciones normalizadas, por ejemplo, por combinación lineal.

De estos pasos, los dos primeros se pueden realizar *offline*.

Esta propuesta es una alternativa a los métodos de normalización que hemos comentado, ya que se mejoran los resultados encontrados en la literatura (cuando se utiliza junto con CombSUM o CombMNZ en la fase de combinación) y evita distorsiones debidas al sesgo. Además, se puede extender, como se indica en [6], empleando datos históricos.

En particular, se indica que las distribuciones F_τ se aproximan a partir de los histogramas de las puntuaciones para cada τ, y $\bar{F}$ se encuentra normalizando primero todos los datos al intervalo $[0, 1]$ linealmente y aproximando el histograma resultante.

3.2. Métodos de combinación

Al igual que en el caso anterior, se pueden distinguir dos tipos de métodos de combinación según utilicen la posición del documento o su puntuación.

Basados en puntuación

$$
\begin{aligned}
\text{CombMIN} &: s_{\mathcal{R}}(d) = \min_{\tau \in \mathcal{R}} \bar{s}_\tau(d) \\
\text{CombMED} &: s_{\mathcal{R}}(d) = \text{median}_{\tau \in \mathcal{R}} \bar{s}_\tau(d) \\
\text{CombMAX} &: s_{\mathcal{R}}(d) = \max_{\tau \in \mathcal{R}} \bar{s}_\tau(d) \\
\text{CombSUM} &: s_{\mathcal{R}}(d) = \sum_{\tau \in \mathcal{R}} \bar{s}_\tau(d) \\
\text{CombANZ} &: s_{\mathcal{R}}(d) = \frac{1}{h(d,\mathcal{R})} \sum_{\tau \in \mathcal{R}} \bar{s}_\tau(d), \\
&\quad\ h(d,\mathcal{R}) = \text{ número de fuentes que recuperan } d \\
\text{CombMNZ} &: s_{\mathcal{R}}(d) = h(d,\mathcal{R}) \sum_{\tau \in \mathcal{R}} \bar{s}_\tau(d)
\end{aligned}
$$

Ninguno de estos métodos precisa de datos de entrenamiento, a pesar de ello, son los más efectivos y, por ello, los más populares métodos de combinación. Entre ellos destacan CombMNZ y CombSUM ([10]), los cuales fueron propuestos por Fox y Shaw ([16]). Además de estos métodos, existen otros dos que sí necesitan datos de entrenamiento: Bartell (que propone diferentes formas de ajustar los pesos, como el método del gradiente conjugado [2]) y Vogt (que utiliza métodos de fusión de redes neuronales [18]).

Basados en posición Los dos métodos que pertenecen a este subgrupo no necesitan datos de entrenamiento, uno de ellos emplea cadenas de Markov, y el otro es el método de Borda ponderado.

Si se usan cadenas de Markov [4] se tiene un conjunto de estados Ω_R que está formado por el conjunto de documentos, además las probabilidades de transición vienen dadas por los rankings devueltos, de manera que podemos definir cuatro modelos dependiendo de cómo se defina la matriz de transiciones, es decir, dado un estado actual $d \in \Omega_R$:

- MC_1: desde el estado actual d se elige uniformemente (al azar) un estado d' tal que $\tau(d') \geq \tau(d)$ para algún τ
- MC_2: primero se elige uniformemente un ranking τ que incluya a d, y después un estado d' en τ tal que $\tau(d') \geq \tau(d)$
- MC_3: igual que el anterior pero eligiendo el estado d' al azar, de manera que si se cumple $\tau(d') \geq \tau(d)$ se realiza la transición a d', y en otro caso nos quedamos en d
- MC_4: se elige un estado d' al azar, si $\tau(d') \geq \tau(d)$ para la mayoría de los $\tau \in R$ se produce la transición a d', si no, seguimos en d

El modelo se utiliza de la siguiente manera: el ranking resultante viene determinado por los valores de probabilidad para cada estado en una distribución estacionaria de la cadena de Markov (es decir, si $P : \Omega_R \rightarrow [0, 1]$

es una distribución estacionaria entonces $s_R(d) = P(d)$). Entre los cuatro modelos presentados los dos que suelen dar mejores resultados son MC_1 y MC_4.

El método de Borda ponderado se basa en el método de Borda que comentamos anteriormente pero en el que cada fuente (τ) tiene una importancia asociada (calidad), con lo que se incluyen estos pesos en la fórmula anterior ([1]).

Métodos híbridos En este caso sólo tenemos el método de la regresión logística, propuesta por Savoy[14] donde la puntuación y la posición se combinan en un modelo de regresión logístico, creando un método híbrido entre los métodos basados en puntuación y los basados en posición. La fórmula es:

$$\begin{aligned} s_R(d) &= \frac{1}{1+e^{-\alpha-\beta\cdot u(d)}} \\ \beta\cdot u(d) &= \sum_{\tau\in R}\beta_{\tau,1}\cdot\tau(d)+\beta_{\tau,2}\cdot s_\tau(d)+\beta_{\tau,3}\cdot\sigma_\tau^2(d) \end{aligned}$$

Donde $\alpha, \beta_{\tau,i}, i = 1, 2, 3$ son parámetros a aprender para cada τ y σ_τ^2 es la varianza de las puntuaciones normalizadas de cada τ.

Hay que decir que se pueden combinar los métodos de normalización basados en posición con los de combinación basados en puntuación; por ejemplo, la normalización de Bayes (posición) seguido de un método de combinación como CombSUM (puntuación) da muy buenos resultados.

3.3. Fusión de rankings usando estimación no parámetrica

En este trabajo proponemos la utilización de la estimación no parámetrica de densidades expuesta en la sección 3.1.1 en el paso de la normalización de puntuaciones, más concretamente, proponemos una mejora al trabajo realizado por Fernández et al. en [5] de la manera que se explica a continuación. Hay que decir que estas técnicas se han aplicado en el campo de la biometría en sistemas multimodales [9], donde para mejorar la calidad de los resultados se disponen de varias fuentes de entrada; se ha observado que funcionan bastante bien, siendo el único límite los datos de entrenamiento disponibles, además resuelve el problema de estimar por distribuciones gaussianas o exponenciales (para documentos relevantes y no relevantes, en nuestro caso, huellas dactilares auténticas e impostoras, en el otro) cuando en realidad los datos no siguen dichas distribuciones (como ocurre por ejemplo en detección de caras).

Como se ha visto en la sección 3.1.1 [5] mejora la fusión de ranking haciendo intervenir a las distribuciones de cada fuente, calculadas a partir de los histogramas, y una distribución común (e ideal) procedente de un histograma generado con los datos de todas las fuentes normalizadas. Nuestra propuesta modifica tres aspectos del trabajo mencionado:

- Estimar de manera no paramétrica cada una de las distribuciones de las fuentes origen.
- Utilizar un proceso de normalización diferente del estándar antes de generar la distribución común.

- Estimar de manera no paramétrica la distribución común.

Utilizando estas técnicas pretendemos obtener características más detalladas de cada fuente, consiguiendo que la fusión de las listas sea más fiel a la realidad que utilizando histogramas, los cuales no son independientes del origen. La modificación del proceso de normalización se propone para evitar que este proceso sea sensible a algún factor (como ocurre con la normalización estándar, que es la que se realiza en el trabajo).

Se tiene previsto hacer pruebas de esta propuesta, de manera que se estudiará si se mejora:

- El proceso original pero con una normalización ZMUV en lugar de la estándar. Si en este paso se han mejorado los resultados, a partir de este momento se utilizará la normalización mencionada.
- Al estimar únicamente la función de distribución común.
- Al estimar cada una de las fuentes por separado. En este paso esperamos encontrar una mejora de los resultados significativa, no obstante, dependerá en gran medida de la generación de la distribución común, por lo que se deberán estudiar con cuidado los efectos de cada una de las dos acciones y decidir cuáles se modifican y cuáles no.

Para las pruebas se tiene previsto utilizar los mismos datos que se emplearon en [5], junto con los pasos de combinación utilizados en dicho trabajo (es decir: CombSUM y CombMNZ). Con respecto al tipo de estimación comenzaremos probando estimadores Kernel uniforme usando la técnica del WARPing, y en virtud al teorema mencionado en la sección 2.1.3 podremos basar todo el análisis utilizando sólo dicho Kernel; sin embargo, y gracias a dicho teorema, si por problemas de precisión el Kernel uniforme no es muy adecuado (por ejemplo, la ventana necesaria es muy pequeña) se podría cambiar de Kernel sin problemas.

No obstante, hay dos aspectos que han quedado sin tratar en este tabajo y que son importantes abarcarlos antes de realizar las pruebas que acabamos de comentar: cómo estimar el ancho de banda (h) y el factor de suavizado (M) óptimos. Esto queda como trabajo futuro (en [15] y [17] se dan algunas ideas al respecto).

4. Conclusiones

En este trabajo se han presentado distintas técnicas de estimación de densidad no paramétrica, se han explicado las distintas fases de la fusión de rankings y se ha propuesto una aplicación de lo primero en determinadas fases de lo segundo. No se disponen aún de pruebas que confirmen la validez de la propuesta, pero se tiene pensado realizarlas lo antes posible. Como se ha visto en este trabajo, la normalización probabilística de las puntuaciones estimando las distribuciones con métodos imperfectos, como los histogramas, ha dado buenos resultados, por lo que si se mejoran dichos métodos, cabe esperar que el proceso entero se perfeccione, y por lo tanto, que se mejoren los resultados.

Referencias

[1] Javed A. Aslam and Mark Montague. Models for metasearch. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 276–284, New York, NY, USA, 2001. ACM.

[2] B. Bartell. *Optimizing Ranking Functions: A Connectionist Approach to Adaptive Information Retrieval*. PhD thesis, 1994.

[3] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, November 2000.

[4] Cynthia Dwork, Ravi S. Kumar, Moni Naor, and D. Sivakumar. Rank aggregation methods for the web. In *World Wide Web*, pages 613–622, 2001.

[5] Miriam Fernández, David Vallet, and Pablo Castells. Probabilistic score normalization for rank aggregation. In *28th European Conference on Information Retrieval (ECIR 2006)*, pages 553–556. Springer Verlag Lecture Notes in Computer Science, Vol. 3936, April 2006.

[6] Miriam Fernández, David Vallet, and Pablo Castells. Using historical data to enhance rank aggregation. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 643–644, New York, NY, USA, August 2006. ACM Press.

[7] Wolfgang Härdle. *Smoothing Techniques: With Implementation in S (Springer Series in Statistics)*. Springer, December 1990.

[8] Wolfgang Härdle, Sigbert Klinke, and Marlene Müller. Applied nonparametric smoothing techniques. Technical report, Humboldt Universitaet Berlin.

[9] Anil Jain, Karthik Nandakumar, and Arun Ross. Score normalization in multimodal biometric systems. *Pattern Recognition*, 38(12):2270–2285, December 2005.

[10] Joon H. Lee. Analyses of multiple evidence combination. In *SIGIR '97: Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 267–276, New York, NY, USA, 1997. ACM Press.

[11] R. Manmatha and H. Sever. A formal approach to score normalization for meta search. In *Human Language Technology Conference (HLT 2002)*, pages 88–93, 2002.

[12] Mark Montague and Javed A. Aslam. Relevance score normalization for metasearch. In *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*, pages 427–433, New York, NY, USA, 2001. ACM.

[13] Arun Ross, Anil K. Jain, and Jian Z. Qian. Information fusion in biometrics. *Lecture Notes in Computer Science (A longer version appears in Pattern Recognition Letters, Vol. 24, Issue 13, pp. 2115-2125, September, 2003.*, 2091:354–359, 2001.

[14] J. Savoy, A. Le Calve, and D. Vrajitoru. Report on the trec-5 experiment: Data fusion and collection fusion, 1988.

[15] David W. Scott. *Multivariate Density Estimation: Theory, Practice, and Visualization (Wiley Series in Probability and Statistics)*. Wiley-Interscience, September 1992.

[16] Joseph A. Shaw and Edward A. Fox. Combination of multiple searches. In *Text REtrieval Conference*, 1994.

[17] B. Turlach. Bandwidth selection in kernel density estimation: A review.

[18] Christopher C. Vogt and Garrison W. Cottrell. Fusion via a linear combination of scores. *Information Retrieval*, 1(3):151–173, 1999.