Selección Dinámica de Índices Métricos para Consultas de Proximidad *

Edgar Chávez
Escuela de Ciencias Físico-Matemáticas
Universidad Michoacana
Morelia - México
elchavez@fismat.umich.mx

Norma Edith Herrera
Departamento de Informática
Universidad Nacional de San Luis
Argentina
nherrera@unsl.edu.ar

Resumen

El modelo de Espacios Métricos permite formalizar el concepto de búsqueda por similitud en bases de datos no tradicionales. El objetivo es construir *estructuras de datos o índices* que permitan reducir el tiempo necesario para resolver una búsqueda por similitud. Uno de los enfoques para la construcción de índices es el usado por los algoritmos basados en pivotes. En este trabajo abordamos el estudio de este grupo de algoritmos, enfocándonos en el diseño de heurísticas para la selección dinámica de un buen grupo de pivotes, y por lo tanto de un buen índice. Para ello, en lugar de construir un único índice para resolver todas las búsquedas, construimos varios índices sobre el espacio con distintos grupos de pivotes y elegimos durante la búsqueda aquel índice que sea más adecuado.

Hemos desarrollado y evaluado varias heurísticas que permiten realizar esta selección, las que han mostrado experimentalmente ser competitivas. La aplicación mas importante de esta técnica es la paralelización de las consultas, si mantenemos cada uno de los índices creados en distintas máquinas de una red.

Palabras claves: Bases de Datos, Espacios Métricos, Indices, Selección de Pivotes

1. Introducción

Nos interesa estudiar el problema de buscar con tolerancia en conjuntos de objetos. Dicho de otra manera, las búsquedas en donde se puedan recuperar objetos *similares* a uno dado. Este tipo de búsqueda se conoce con el nombre de ***búsqueda por proximidad o búsqueda por similitud***, y surge en diversas áreas tales como reconocimiento de voz, reconocimiento de imágenes, compresión de texto, biología computacional, inteligencia artificial, minería de datos, entre otras.

Para lo anterior abstraemos el problema a un universo de objetos $\mathcal{X}$ y una función de distancia d que modela la similitud entre los objetos del universo. Esta función d cumple con las propiedades características de una función de distancia: *positividad* ($d(x, y) \geq 0$), *simetría* ($d(x, y) = d(y, x)$) y *desigualdad triangular* ($d(x, y) \leq d(x, z) + d(z, y)$).

El par $(\mathcal{X}, d)$ se denomina ***espacio métrico***. La base de datos será un subconjunto finito $\mathcal{U} \subseteq \mathcal{X}$. En este nuevo modelo de bases de datos, una de las consultas típicas que implica recuperar objetos

*Este trabajo ha sido parcialmente subvencionado por CYTED VII.19 RIBIDI Project, por el proyecto CONACyT 36911A, y por el proyecto 22/F314 (UNSL)

similares es la *búsqueda por rango*, que denotaremos con $(q, r)_d$. Dado un elemento $q \in \mathcal{X}$, al que llamaremos *query* y un radio de tolerancia r, una búsqueda por rango consiste en recuperar los objetos de la base de datos cuya distancia a q no sea mayor que r, es decir, $(q, r)_d = \{u \in \mathcal{U} : d(q, u) \leq r\}$.

El tiempo total de resolución de una búsqueda contiene tres términos, a saber: *T= #evaluaciones de* $d\times$ *complejidad(d) + tiempo extra de CPU + tiempo de I/O* En muchas aplicaciones la evaluación de la función d es tan costosa que las demás componentes de la fórmula anterior pueden ser despreciadas. Éste es el modelo usado en este trabajo; por consiguiente, nuestra medida de complejidad será la cantidad de evaluaciones de la función de distancia d.

Una forma trivial de resolver una búsqueda por rango es examinando exhaustivamente la base de datos. Para evitar esta situación, se preprocesa la base de datos por medio de un *algoritmo de indización* con el objetivo de construir una *estructura de datos o índice*, diseñada para ahorrar cálculos en el momento de resolver una búsqueda. En [4] se presenta un desarrollo unificador de las soluciones existentes en la temática. En dicho trabajo se muestra que existen dos grupos de algoritmos de indización: *algoritmos basados en pivotes* y *algoritmos basados en particiones compactas*.

Uno de los principales obstáculos en el diseño de buenas técnicas de indización es lo que se conoce con el nombre de *maldición de la dimensionalidad*. El concepto de dimensionalidad está relacionado a la dificultad o facilidad de buscar en un determinado espacio métrico. La dimensión intrínseca de un espacio métrico se define en [4] como $\rho = \frac{\mu^2}{2\sigma^2}$, siendo μ y σ^2 la media y la varianza respectivamente de su histograma de distancias. Es decir que, a medida que la dimensionalidad intrínseca crece, la media crece y su varianza se reduce. Esto significa que el histograma de distancia se concentra más alrededor de su media, lo que influye negativamente en los algoritmos de indización.

En este trabajo abordamos el estudio de algoritmos de indización basados en pivotes, enfocándonos en el estudio de técnicas para la selección dinámica de un buen grupo de pivotes, y por lo tanto de un buen índice. Para ello, en lugar de construir un único índice para resolver todas las búsquedas, construimos varios índices sobre el espacio con distintos grupos de pivotes y elegimos durante la búsqueda aquel índice que sea más adecuado. Hemos desarrollado y evaluado varias técnicas que permiten realizar esta selección. La aplicación mas importante de esta técnica es la paralelización de las consultas, si mantenemos cada uno de los índices creados en distintas máquinas de una red.

Comenzamos dando una breve explicación sobre algoritmos basados en pivotes, para luego presentar las heirísticas de selección dinámica de índices. Luego mostramos la evaluación experimental de nuestra propuesta, que nos permitió además encontrar caracterizaciones de buenos grupos de pivotes. Finalizamos dando las conclusiones y el trabajo futuro.

2. Algoritmos Basados en Pivotes

Este grupo de algoritmos construyen el índice basándose en la distancia de los objetos de la base de datos a un conjunto de elementos preseleccionados que llamaremos *pivotes*.

Para ello, se seleccionan k pivotes $\{p_1, p_2, \ldots, p_k\}$, y se le asigna a cada elemento a de la base de datos, el vector o firma $\Phi(a) = (d(a, p_1), d(a, p_2), \ldots, d(a, p_k))$. Durante la búsqueda se usa la desigualdad triangular junto con la firma de cada elemento para filtrar objetos de la base de datos sin medir su distancia a la query q. Dada una búsqueda $(q, r)_d$, se computa la firma de la query q, $\Phi(q) = (d(q, p_1), d(q, p_2), \ldots, d(q, p_k))$, y luego se descartan todos aquellos elementos a, tales que para algún pivote p_i se cumple que $\mid d(q, p_i) - d(a, p_i) \mid > r$, es decir: $max_{1 \leq i \leq k}\{\mid d(a, p_i) - d(q, p_i) \mid \} = L_\infty(\Phi(a), \Phi(q)) \leq r$

Tal como lo mencionáramos en la sección anterior, uno de los principales obstáculos en el diseño de buenas técnicas de indización es lo que se conoce con el nombre de *maldición de la dimensionali-*

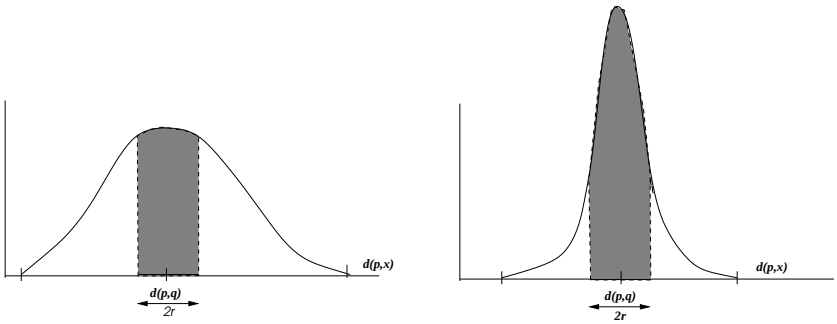

Figura 1: *Histogramas de distancias de baja dimensionalidad (izquierda), y de alta dimensionalidad (derecha)*

dad. La figura 1 da una idea intuitiva de por qué el problema de búsqueda se torna más difícil cuando el histograma del espacio es más concentrado. Consideremos una búsqueda $(q, r)_d$ y un índice basado en pivotes elegidos aleatoriamente. En la figura se ejemplifican dos posibles casos para el histograma local respecto del punto p. Si p es un pivote, estas gráficas representan dos posibles distribuciones para los valores de $d(q, p)$. La regla de eliminación dice que podemos descartar aquellos puntos y tales que $y \notin [d(p, q) - r, d(p, q) + r]$. Las áreas sombreadas muestran los puntos que no podrán descartarse. Esto significa que a medida que el histograma se concentra más alrededor de su media disminuye la cantidad de puntos que pueden descartarse usando como dato $d(p, q)$.

3. Heurísticas para la Selección Dinámica de Índices

Se sabe que la política usada en la selección de pivotes afecta notablemente la performance de la búsqueda [1, 4, 5, 6]. Esto significa que si tenemos dos conjuntos de pivotes del mismo tamaño elegir el mejor de los dos puede reducir el tiempo de búsqueda. Por otro lado, un grupo pequeño de pivotes bien elegidos puede resultar tan eficiente como un grupo de mayor cantidad de pivotes pero elegidos aleatoriamente. En consecuencia, el tema de selección de un buen grupo de pivotes para indizar un determinado espacio métrico está siendo ampliamente estudiado.

Nuestra propuesta aquí es una selección dinámica del conjunto de pivotes, y por consiguiente del índice, sobre el que se resolverá la consulta. Para ello, en lugar de seleccionar durante la construcción del índice un grupo de pivotes que sea efectivo para todo el espacio métrico, *seleccionamos durante la búsqueda un grupo de pivotes que sea efectivo para la query* q. Para ello, construimos varios índices sobre el espacio con distintos grupos de pivotes (elegidos aleatoriamente); luego, durante una búsqueda $(q, r)_d$ seleccionamos aquel índice que sea más adecuado a q de acuerdo al conjunto de pivotes con el que fue construido. Esta selección dinámica de índices permite además realizar consultas en paralelo, si cada uno de los índices creados se mantiene en distintas máquinas de una red.

Supongamos que hemos generado M índices de k pivotes cada uno. Durante una búsqueda seleccionamos uno de ellos para trabajar. Si sólo tomamos en cuenta la cantidad de evaluaciones de distancia, esta idea perderá al compararla con la opción de tener un sólo índice de Mk pivotes. Pero si tomamos en cuenta tiempo extra de CPU y tiempo de I/O, la idea de varios índices pequeños aventaja a la opción de un sólo índice con mayor cantidad de pivotes.

Presentamos a continuación las heurísticas que hemos diseñado para seleccionar el índice más adecuado a una búsqueda $(q, r)_d$. En el desarrollo de la explicación supondremos que hemos generado M índices de k pivotes cada uno. Denotaremos con I_j al índice j, y con p_i^j al pivote i del índice I_j.

3.1. Selección por votos

Tal como se muestra en [2], la efectividad de un pivote depende de su distancia a la query q. Los pivotes más efectivos son los más cercanos o los más alejados a q. En nuestro caso el problema no es seleccionar un pivote sino seleccionar un conjunto de pivotes que sea el más efectivo para una búsqueda $(q, r)_d$. Entonces, basándonos en el criterio anterior, podemos extender esas ideas de la siguiente manera:

- Calcular la distancia de q a todos los pivotes: $(\forall j)_{1 \leq j \leq M}\ (\forall i)_{1 \leq i \leq k}$ calcular $d(q, p_i^j)$
- Ordenar los pivotes de acuerdo a su distancia a q. Sea $S = \langle \mathfrak{p}_1,\ \mathfrak{p}_2,\ \ldots,\ \mathfrak{p}_{M*k} \rangle$ esta secuencia ordenada de pivotes; luego cada elemento $\mathfrak{p}_s$ de S será de la forma $\mathfrak{p}_s = p_i^j$.
- Seleccionar aquel índice que tiene mayor cantidad de pivotes cercanos y lejanos a q. Esto significa que por cada pivote que se encuentra en los extremos de S se agrega un voto al índice al que pertenece ese pivote. El índice con mayor cantidad de votos es el que resulta seleccionado.

Un punto no definido en el proceso anterior es qué tomar como "extremo" de S. Para manejar esta situación utilizaremos un parámetro (e) que delimitará los extremos de S; es decir, de la secuencia S sólo votarán los siguientes pivotes: $\mathfrak{p}_1,\ \mathfrak{p}_2,\ \ldots,\ \mathfrak{p}_e,\ \mathfrak{p}_{M*k-e},\ \ldots,\ \mathfrak{p}_{M*k-1},\ \mathfrak{p}_{M*k}$

3.2. El pivote más cercano

Bajo el mismo criterio de la política anterior, otra posibilidad es seleccionar aquel índice que tenga el pivote más cercano a la query q:

- $(\forall j)_{1 \leq j \leq M}\ (\forall i)_{1 \leq i \leq k}$: calcular $d(q, p_i^j)$
- Obtener $S = \langle \mathfrak{p}_1,\ \mathfrak{p}_2,\ \ldots,\ \mathfrak{p}_{M*k} \rangle$ la secuencia ordenada de pivotes según su distancia a q.
- Si $\mathfrak{p}_1 = p_i^j$, seleccionar el índice I_j.

3.3. El pivote más lejano

Otra alternativa es seleccionar aquel índice que tenga el pivote más lejano a la query q; para esto, procedemos como en el caso anterior para obtener la secuencia S y luego, si $\mathfrak{p}_{M*k} = p_i^j$, seleccionamos el índice I_j.

3.4. Menor masa total

Dado un pivote p y una búsqueda $(q, r)_d$ sabemos que los elementos que no pueden ser eliminados por p son aquellos x tales que $d(x, p) \in [d(p, q) - r, d(p, q) + r]$. En la figura 1 la cantidad de elementos que no podrán eliminarse ante una búsqueda $(q, r)_d$, corresponde al área sombreada del histograma. Llamaremos a esta zona la *masa* de p para la query q, y la denotaremos con $m(p, q)$.

Utilizando este hecho, una de las políticas experimentadas fue la de seleccionar aquel índice que tenga la menor masa total, siendo la masa total de un índice la suma de las masas de todos sus pivotes. Podemos resumir esto en los siguiente pasos:

- $(\forall j)_{1 \leq j \leq M}$ calcular $m(I_j, q) = \sum_{i=1}^{k} m(p_i^j, q)$
- Seleccionar aquel índice I_j tal que: $(\forall l)_{1 \leq l \leq M}\ m(I_j, q) \leq m(I_l, q)$.

Dado que durante la construcción de un índice se calculan las distancia de todos los elementos de la base de datos a todos los pivotes, en ese momento es posible obtener el histograma de los pivotes sin costo adicional.

3.5. Pivote de menor masa

Siguiendo con el planteamiento del punto anterior también se experimentó seleccionar aquel índice que tuviera el pivote con menor masa:

- $(\forall j)_{1 \leq j \leq M}\ (\forall i)_{1 \leq i \leq k}$: calcular $m(p_i^j, q)$
- Seleccionar aquel índice I_j tal que $m(p_i^j, q)$ sea mínima para algún i, con $1 \leq i \leq k$.

3.6. Una combinación de todas las técnicas: votación global

Las técnicas descriptas en los puntos anteriores se basan en visiones diferentes del espacio métrico en base a las cuales toman sus decisiones. Cada una de las ellas puede no sólo seleccionar un índice sino ordenar todos los índices según la conveniencia para una búsqueda $(q, r)_d$. Obviamente cada una de las políticas establecidas puede cometer errores; en estos casos el índice óptimo para la query q queda desplazado del primer lugar.

Una forma de combinar las técnicas con el objetivo de disminuir los errores que pueden ocurrir en la elección es asignando votos a los índices según la técnica que lo seleccionó. Sea $\langle I_{j_1}, I_{j_2}, \ldots, I_{j_M} \rangle$ la secuencia de índices ordenados por alguna técnica de acuerdo a su conveniencia para la query q. Luego, I_{j_s} recibe una cantidad de votos que depende de la probabilidad de que la técnica considerada desplace a la posición s el índice óptimo para q. Estos valores se establecieron experimentalmente, y se explican en la siguiente sección.

4. Evaluación Experimental de las Políticas de Selección

4.1. Descripción de los experimentos

Los experimentos fueron realizados usando como espacio métrico un diccionario español de 86.061 palabras y usando como función de distancia la distancia de edición. Esta función es discreta y calcula la mínima cantidad de palabras que hay que agregar, cambiar y/o eliminar a una palabra para obtener otra. Este modelo es comúnmente usado en recuperación de texto, procesamiento de señales y aplicaciones de biología computacional. En cuanto al algoritmo de indización se utilizó el *Fixed Queries Trie (FQTrie)* [3] para indizar el diccionario español.

Cada una de las técnicas descriptas anteriormente fue probada con índices de 8 pivotes, tomando grupos de 10, 20 y 30 índices sobre el diccionario español; es decir $k = 8$ y $M = 10, 20, 30$. Cabe señalar que el grupo de 20 índices se armó usando los mismos 10 índices generados para $M = 10$ más 10 índices nuevos; y el grupo de 30 índices se armó utilizando los 20 índices anteriores más 10 índices nuevos. Esto permitió evaluar qué efecto tiene, sobre las distintas técnicas, aumentar la cantidad de índices sobre las que trabajan.

Se eligieron al azar 500 palabras del diccionario y para cada una de ellas se realizaron búsquedas por rango $(q, r)_d$ usando como radio de búsqueda r los valores $1,\ 2,\ 3$ y 4. Cada una de estas búsquedas fue realizada en cada uno de los índices. Esto nos permitió obtener para cada $(q, r)_d$ cuál era el índice óptimo (es decir el que realiza la menor cantidad de evaluaciones de distancia). A partir de esta información se pudo evaluar la bondad de cada una de las políticas de selección implementadas.

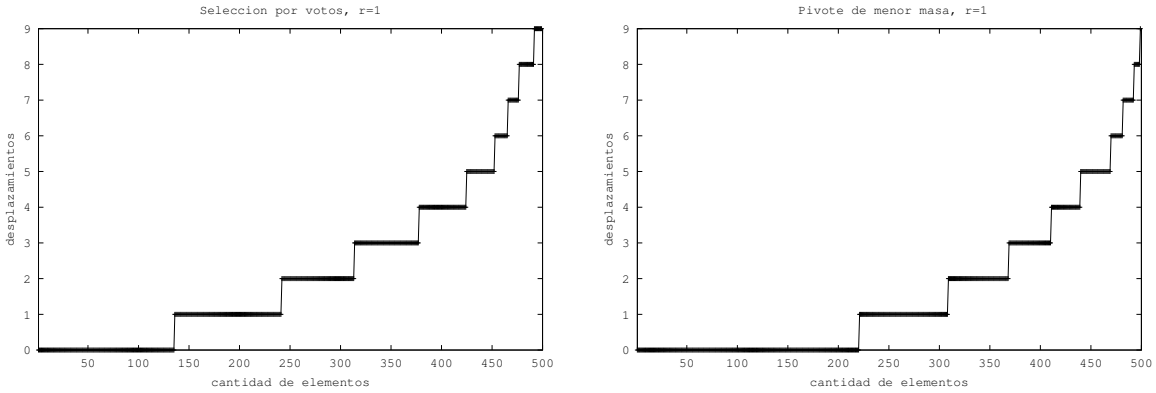

Figura 2: *Desplazamiento del índice óptimo para selección por votos y pivote de menor masa, para* $r = 1$ *y* $M = 10$.

En el caso de la selección por votos para el parámetro e se usaron los valores $k/2, k, 2k$ y $3k$, observándose el mejor desempeño para $e = k/2$. En consecuencia, ese fue el valor utilizado en todos los experimentos.

Para cada una de las técnicas se analizaron los siguientes aspectos:

- Cada una de las técnicas implementadas, excepto la votación global, retorna no sólo el índice seleccionado, sino todos los índices ordenados por conveniencia de acuerdo a su política. De esta manera, podemos analizar cuánto se desplaza el índice que debería haber sido seleccionado respecto del que resulta electo.

- Porcentaje de fallas, es decir, la cantidad de veces que se equivoca en la elección del mejor índice para una búsqueda $(q, r)_d$.

- Para cada una de las búsquedas se calculó la proporción $cmpe/cmpm$, siendo $cmpe$ la cantidad de comparaciones hechas para buscar en el índice que resulta elegido y $cmpm$ la cantidad de comparaciones realizadas si se busca en el mejor índice para la query q. Esto permite evaluar cuánto empeora la técnica respecto del ideal (no fallar nunca en la elección).

- Para cada una de las búsquedas también se calculó la proporción $cmpe/cmpa$, siendo $cmpe$ el definido anteriormente, y $cmpa$ la cantidad de comparaciones hechas si se busca en un índice elegido aleatoriamente. Esto permite evaluar si la técnica mejora respecto de una selección totalmente aleatoria.

Por cuestiones de espacio, en la próxima sección sólo mostramos las gráficas de los resultados que consideramos más significativos.

4.2. Análisis de Resultados

Comenzamos los experimentos realizando el análisis del desplazamiento del óptimo para cada una de las técnicas propuestas. La figura 2 ilustra este comportamiento para algunas de las técnicas de selección, usando como radio de búsqueda $r = 1$. Se pudo observar que las técnicas basadas en cálculos de masa son las que tienen mayor porcentaje de éxito y un menor desplazamiento del óptimo; en alrededor del 50 % de los casos el óptimo se encuentra con un desplazamiento de 0 o 1. En general, aumentar el radio de búsqueda mejora la predicción del mejor índice en todas las técnicas, pero esta mejora es más significativa en el caso de las técnicas basadas en cálculo de masas.

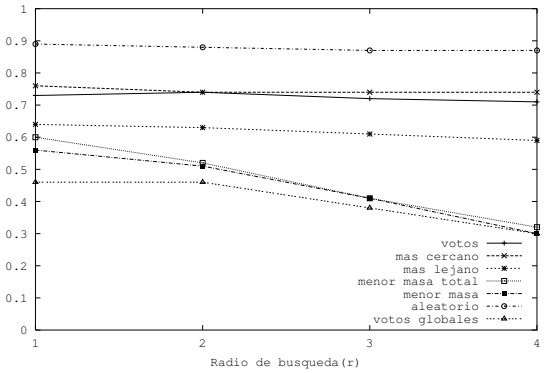

Figura 3: *Porcentaje de fallas, para un lote de prueba de 500 palabras, y* $M = 10$.

En base a estos resultados, para la técnica de votación global solamente se utilizaron los índices que se encuentran con desplazamientos 0 o 1. La cantidad de votos que cada técnica asigna se estableció en función de la probabilidad de éxito. Por ejemplo, la técnica de selección por el pivote más lejano encuentra el mejor índice (desplazamiento 0) en el 35 % de los casos; y en el 19 % de los casos el óptimo se desplaza un lugar. En consecuencia, esta técnica asigna 3 votos al índice que selecciona con desplazamiento 0 y 2 votos al que selecciona con desplazamiento 1.

La figura 3, muestra el porcentaje de fallas de cada una de las técnicas. En este sentido, la de mejor desempeño es la votación global seleccionando el índice óptimo en el 55 % de los casos o más dependiendo del radio de búsqueda; le siguen la técnica de menor masa y luego menor masa total. Nuestra intuición respecto de esta observación es que la votación global logra una visión más amplia del espacio a partir de la información provista por las restantes técnicas. En base a los resultados de los experimentos, pudimos observar que generalmente donde alguna técnica falla otras aciertan en la elección. Esta información es captada por la votación global posibilitando así la corrección de errores.

La figura 4 (izquierda) muestra el promedio de las proporciones $cmpe/cmpm$ sobre las 500 búsquedas realizadas. Nuevamente, las de mejor desempeño son votación global, menor masa y menor masa total. Sin embargo, en este caso la diferencia entre menor masa y votación global es pequeña. Esto significa que si bien menor masa se equivoca en la elección un 10 % más que votación global (ver figura 3) el índice que resulta electo no es mucho peor que el elegido por votación global. No sucede lo mismo con menor masa total. En este caso el error es un 15 % más que votación global pero, cuando se equivoca, el índice que resulta electo tiene un desempeño notablemente inferior al índice

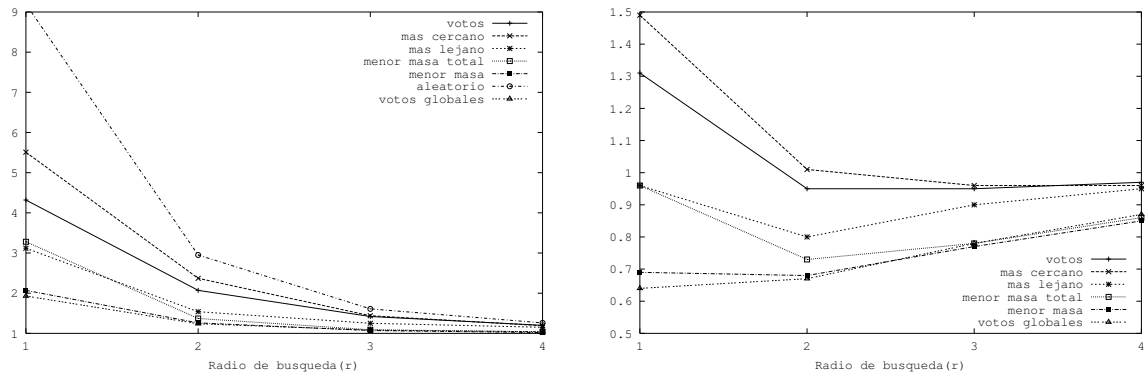

Figura 4: *Promedio de las proporciones* $cmpe/cmpm$ *(izquierda) y* $cmpe/cmpa$ *(derecha), para* $M = 10$.

elegido por votación global.

La figura 4 (derecha) muestra el promedio de las proporciones $cmpe/cmpa$ sobre las 500 búsquedas realizadas. Nuevamente las de mejor desempeño son votación global y menor masa. Estas técnicas realizan un 30 % menos de evaluaciones de distancia que una elección completamente al azar. Notar que selección por votos y más cercano realizan en promedio más evaluaciones de distancia que una selección aleatoria. Estas técnicas, si bién comenten menos errores en la predicción que una elección aleatoria, cuando se equivocan eligen un índice marcadamente inferior que una selección al azar.

El próximo paso en nuestros experimentos fue aumentar la cantidad de índices. Con respecto a los porcentajes de fallas, aumentar la cantidad de índices produjo un incremento en los errores que comenten las técnicas (ver fig. 5); se mantiene que las de mejor desempeño son pivote de menor masa y votación global.

Las figuras 6 y 7 muestran el promedio de las proporciones $cmpe/cmpm$ y $cmpe/cmpa$, para $M = 20$ y $M = 30$ respectivamente. Nuevamente, pivote de menor masa y votación global son las de mejor desempeño y las que muestran un comportamiento más estable ante el aumento de la cantidad de índices sobre los que trabajan. Esto no sucede con las restantes técnicas: incrementar la cantidad de índices influye negativamente en el desempeño de selección por votos, más cercano, más lejano y menor masa total.

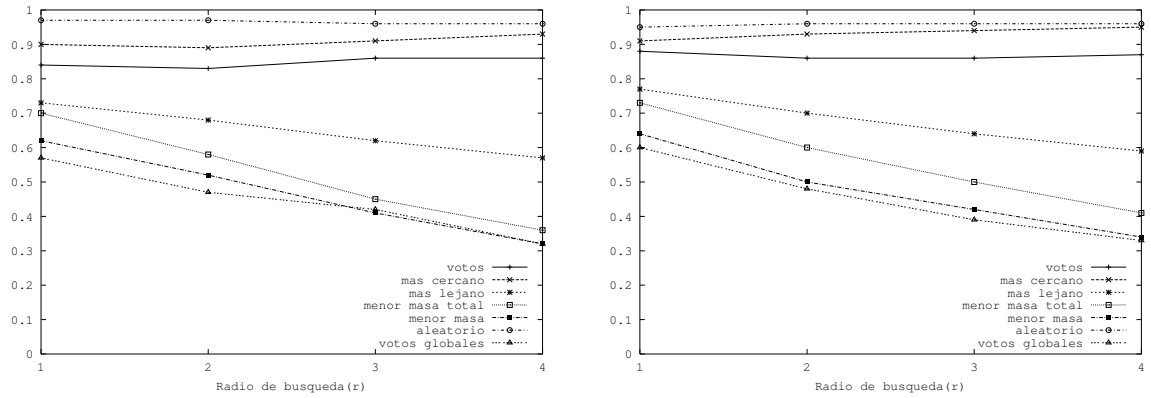

Figura 5: *Porcentaje de fallas, para $M = 20$ (izquierda) y $M = 30$ (derecha).*

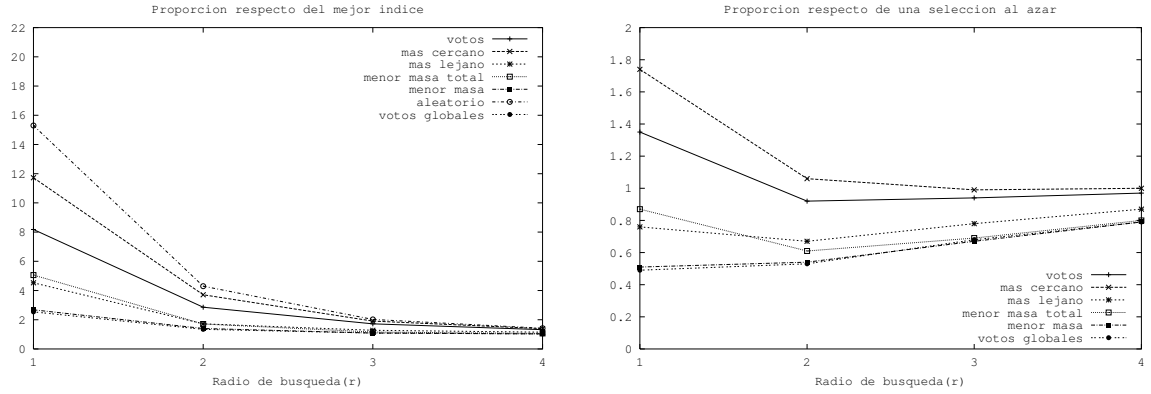

Figura 6: *Promedio de las proporciones $cmpe/cmpm$ (izquierda) y $cmpe/cmpa$ (derecha), para $M = 20$.*

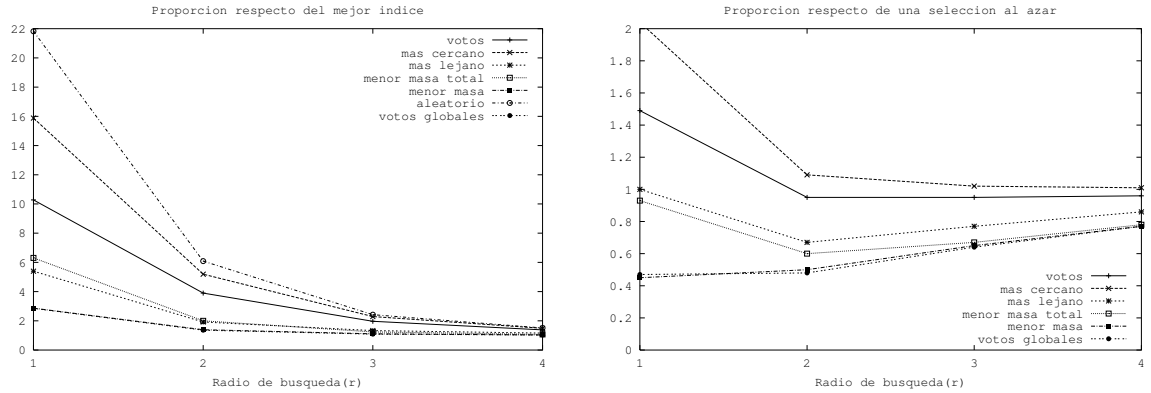

Figura 7: *Promedio de las proporciones* $cmpe/cmpm$ *(izquierda) y* $cmpe/cmpa$ *(derecha), para* $M = 30$.

5. Propiedades que caracterizan a un buen grupo de pivotes

Los experimentos que expusimos en la sección anterior nos permitieron no sólo evaluar el comportamiento de las distintas políticas de selección propuestas sino también analizar las características que presentaba cada uno de los grupos de pivotes utilizados en la construcción de los índices.

El cuadro 1 muestra, para el caso de $M = 10$, la cantidad de veces que cada uno de los índices resulta ser el óptimo para una query q, en función de los distintos radios de búsqueda. Dentro de cada columna, se han ordenado los índices por orden creciente de cantidad. Como puede observase el índice 3 es el que mayor cantidad de veces resulta ser el óptimo, independientemente del radio de búsqueda; y el índice 0 es el que menor cantidad de veces es el óptimo, también independientemente del radio de búsqueda. Para los restantes índices las cantidades varían dependiendo de r. Los que presentan mayor variación son los índices 2 y 4. En el caso del índice 2, para $r = 1$ es el mejor en 48 casos, y para $r = 4$ esta cantidad disminuye a 17. En el caso del índice 4, para $r = 1$ es el óptimo en 44 casos, y para $r = 4$ esta cantidad aumenta a 100.

La figura 8 muestra los histogramas de distancias de los pivotes usados en los índices $3, 0, 4$ y 2, del grupo de 10 índices. En el caso del índice 3 es donde se puede observar una mayor variación de histogramas. En el caso del índice 0 los histogramas de los primeros pivotes son muy similares y esto es lo que puede provocar el bajo desempeño del mismo.

Este comportamiento se repite para el grupo de 20 índices. El cuadro 2 muestra la cantidad de veces que cada índice resulta ser el óptimo para una búsqueda; por cuestiones de espacio sólo se dan los peores y mejores casos para cada radio de búsqueda. Notar que nuevamente aparece el índice 3 entre los mejores y el índice 0 entre los peores. Los histogramas de estos índices se muestran en la figura 9. Notar la similitud entre los histogramas de los pivotes del índice 17. Esto significa que todos

$r = 1$	$r = 2$	$r = 3$	$r = 4$
índice 0 = 32	índice 0 = 25	índice 0 = 17	índice 0 = 12
índice 7 = 34	índice 9 = 28	índice 5 = 24	índice 2 = 17
índice 1 = 35	índice 7 = 29	índice 2 = 27	índice 5 = 17
índice 9 = 37	índice 1 = 30	índice 7 = 27	índice 7 = 31
índice 5 = 40	índice 5 = 36	índice 1 = 34	índice 9 = 34
índice 6 = 43	índice 2 = 40	índice 9 = 38	índice 1 = 35
índice 4 = 44	índice 6 = 41	índice 6 = 43	índice 6 = 40
índice 2 = 48	índice 4 = 64	índice 8 = 75	índice 8 = 78
índice 8 = 71	índice 8 = 70	índice 4 = 82	índice 4 = 100
índice 3 = 116	índice 3 = 137	índice 3 = 133	índice 3 = 136

Cuadro 1: *Cantidad de veces que cada índice resulta ser el óptimo sobre un total de 500 búsquedas, para* $M = 10$.

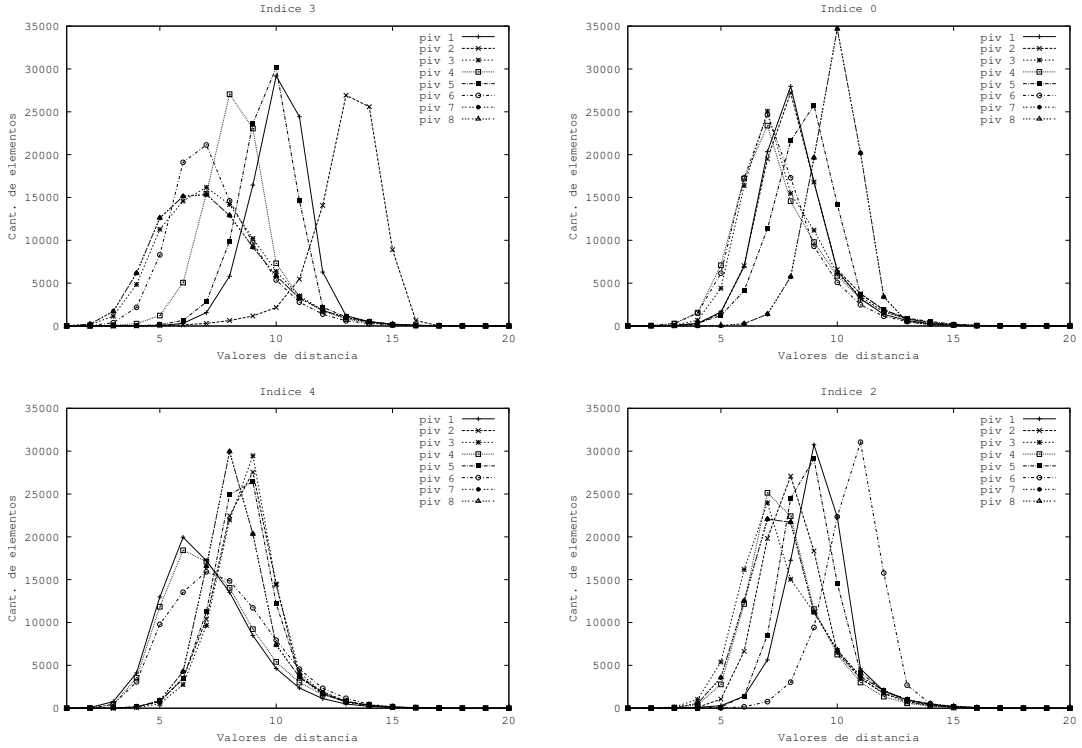

Figura 8: *Histograma de distancia de los pivotes usados en la creación de los índices 3, 0, 4 y 2, del grupo* $M = 10$.

$r = 1$	$r = 2$	$r = 3$	$r = 4$
indice 7 = 11	indice 0 = 7	indice 0 = 8	indice 17 = 2
indice 9 = 14	indice 5 = 10	indice 16 = 9	indice 16 = 4
⋮	⋮	⋮	⋮
indice 3 = 46	indice 19 = 50	indice 8 = 60	indice 4 = 66
indice 19 = 62	indice 3 = 65	indice 3 = 69	indice 3 = 79

Cuadro 2: *Cantidad de veces que cada índice resulta ser el óptimo sobre un total de 500 búsquedas, para* $M = 20$.

los pivotes de ese índice tienen la misma perspectiva del espacio y, en consecuencia, los grupos de elementos que cada uno de ellos descarta son muy similares.

El cuadro 3 muestra para cada índice la cantidad de veces que es el óptimo en búsquedas de radio $r = 1$, y la media y varianza de las distancias del conjunto de pivotes. Se han ordenado los índices por orden creciente de cantidad. Los índices más competitivos son, en general, los de mayor varianza; notar que el histograma del pivote 1 del índice 3 es justamente el que más alejado se encuentra del resto (es un outlier). En el caso del índice 13, si bién su varianza no es una de las mayores, figura entre los más elegidos; y el índice 5, a pesar de tener una varianza de 4.17, no está dentro del grupo de los más competitivos. La razón de estos comportamientos pueden encontrarse en los histogramas de distancias de los pivotes de estos índices (ver figura 10). Se puede observar que el índice 13 tiene un pivote cuyo histograma se corresponde con un espacio de baja dimensionalidad. Esto significa que buenos grupos de pivotes, además de contener un punto que sea outlier, también deben tener un elemento cuyo histograma sea de baja dimensionalidad.

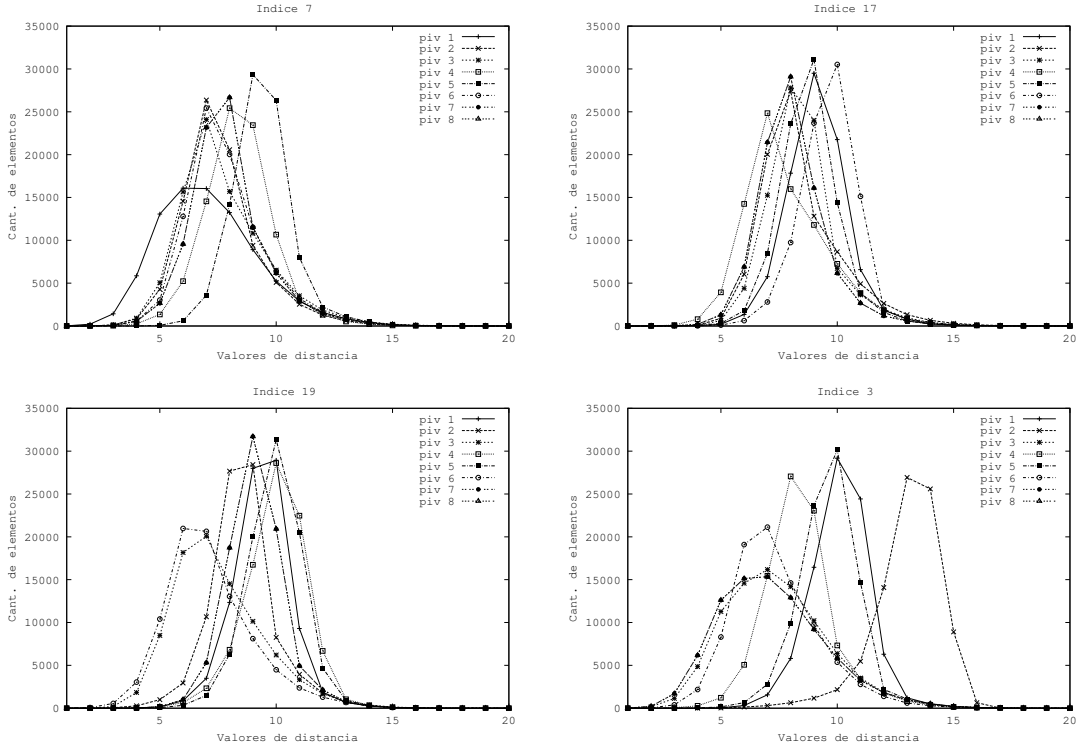

Figura 9: *Histograma de distancia de los pivotes usados en la creación de los índices 7, 17, 19 y 3 del grupo* $M = 20$.

elecciones	media	varianza
indice 7 = 11	7.32	3.43
indice 9 = 14	7.14	3.10
indice 16 = 14	9.04	3.35
indice 0 = 15	7.68	3.28
indice 1 = 15	8.39	4.19
indice 10 = 15	8.07	3.39
indice 12 = 17	8.57	3.66
indice 2 = 18	8.64	3.94
indice 17 = 18	8.71	3.36
indice 5 = 19	8.07	4.17

elecciones	media	varianza
indice 6 = 21	7.71	3.60
indice 4 = 22	7.75	3.41
indice 8 = 27	8.36	4.39
indice 15 = 28	8.68	4.25
indice 18 = 28	8.54	3.80
indice 11 = 31	9.25	4.57
indice 13 = 38	8.32	3.79
indice 14 = 41	9.79	5.85
indice 3 = 46	9.29	5.05
indice 19 = 62	9.86	4.35

Cuadro 3: *Media y varianza de distancias de los conjuntos de pivotes*

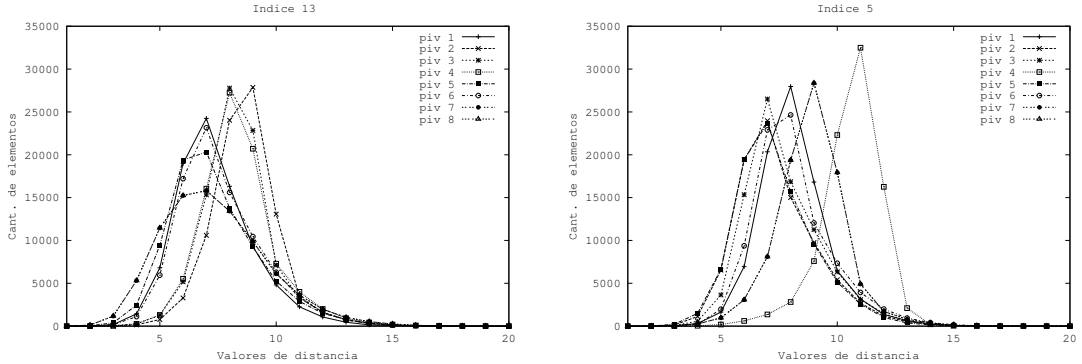

Figura 10: *Histogramas de distancias de los índices 13 y 5*

6. Conclusiones y Trabajo Futuro

En este trabajo hemos presentado heurísticas para la selección dinámica de un buen grupo de pivotes, y por consiguiente de un buen índice. Para ello, construimos varios índices sobre el espacio con distintos grupos de pivotes y, durante una búsqueda $(q, r)_d$, elegimos aquel índice que sea más adecuado para q. La aplicación más importante de esta técnica es la paralelización de las consultas cuando se replica el índice con parámetros distintos (distintos grupos de pivotes) en cada nodo.

Hemos propuesto varias políticas para realizar la selección de un índice adecuado. De todas las políticas presentadas las de mejor desempeño fueron pivote de menor masa y votación global. Los porcentajes de falla de estas técnicas oscilan entre 30 % y el 60 %, dependiendo del radio de búsqueda (r) y la cantidad de índices (M).

Dos políticas demostraron ser peor que una selección aleatoria: selección por votos y pivote más cercano. Estas técnicas, si bién cometen menos errores que una elección aleatoria, cuando se equivocan eligen un índice de desempeño inferior que una selección al azar.

Al analizar las caracteríticas que presentan los pivotes de los índices más competitivos, pudimos concluir que los histogramas de buenos grupos de pivotes presentan una mayor variación. Tal como esperábamos, los índices de mejor desempeño contienen entre sus pivotes un outlier. Pudimos observar además que los mejores índices son aquellos cuyo grupo de pivotes tienen una mayor varianza. Esta característica por sí sola no son suficiente; para obtener un buen desempeño es necesario además que el grupo de pivotes contenga un elemento cuyo histograma sea de baja dimensionalidad.

Algunos puntos interesantes para abordar en futuras investigaciones son los siguientes. Una primera posibilidad es usar las políticas para selección de un índice adecuado a una query q con el objetivo de particionar el espacio. La idea es, en lugar de mantener todos los elementos de la base de datos en todos los índices, insertar cada elemento sólo en el índice más adecuado.

Referencias

[1] B. Bustos, G. Navarro, and E. Chávez. Pivot selection techniques for proximity searching in metric spaces. In *Proc. of the XXI Conference of the Chilean Computer Science Society (SCCC'01)*, pages 33–40. IEEE CS Press, 2001.

[2] Cengiz Celik. Priority vantage points structures for similarity queries in metric spaces. In *EurAsia-ICT 2002*, LCNS 2510, pages 256–263. Springer-Verlag, 2002.

[3] E. Chávez and K. Figueroa. Faster proximity searching in metric data. In *Proceedings of MICAI 2004. LNCS 2972, Springer*, Cd. de México, México, 2004.

[4] E. Chávez, G. Navarro, R. Baeza-Yates, and J.L. Marroquín. Searching in metric spaces. *ACM Computing Surveys*, 33(3):273–321, September 2001.

[5] A. Faragó, T. Linder, and G. Lugosi. Fast nearest-neighbor search in dissimilarity spaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(9):957–962, 1993.

[6] L. Micó, J. Oncina, and E. Vidal. A new version of the nearest-neighbor approximating and eliminating search (AESA) with linear preprocessing-time and memory requirements. *Pattern Recognition Letters*, 15:9–17, 1994.