Avances sobre Bases de Datos no Convencionales

Jorge Arroyuelo, Susana Esquivel, Alejandro Grosso, Verónica Ludueña, Nora Reyes
Dpto. de Informática, Fac. de Cs. Físico-Matemáticas y Naturales, Universidad Nacional de San Luis
{*bjarroyu, esquivel, agrosso, vlud, nreyes*}*@unsl.edu.ar*
Edgar Chávez
Centro de Investigación Científica y de Educación Superior de Ensenada, México
elchavez@cicese.mx
Gonzalo Navarro
Departamento de Ciencias de la Computación, Universidad de Chile.
gnavarro@dcc.uchile.cl

Resumen

En la actualidad han surgido, debido a los avances alcanzados en tecnologías de información y comunicación, aplicaciones no tradicionales sobre bases de datos de todo tipo de datos, valga la redundancia. Éstos pueden ser imágenes, texto libre, secuencias de ADN, audio, video, etc., pueden estar o no estructurados y provenir de diversas fuentes (satélites, telescopios, revistas, fotografías, música, etc.), como también ser de diferentes tamaños, generalmente muy grandes. Por otro lado, las consultas a los mismos, en el sentido clásico, no resultan significativas, todas las consultas, sobre estos datos *multimediales* son por objetos similares a uno dado. Estos escenarios requieren modelos más generales, como las *Bases de Datos Métricas*, pero que logren una madurez semejante al de las bases de datos tradicionales.

Por otra parte, los lenguajes de consulta, necesarios para la administración de una base de datos, no siempre poseen el poder expresivo necesario para expresar todas las consultas consideradas de interés en este modelo. Además, el desarrollo de memorias más rápidas y de gran capacidad, promovió la aparición de estructuras de datos que tienen en cuenta estas arquitecturas como las *estructuras de datos con I/O eficiente*. Nuestra investigación pretende contribuir a la madurez y consolidación de este nuevo modelo de bases de datos.

Palabras Claves: bases de datos no convencionales, lenguajes de consulta, índices, expresividad.

Contexto

La línea *Bases de Datos no Convencionales*, la cual motiva esta presentación, es parte del Proyecto Consolidado 330303 "Tecnologías Avanzadas de Bases de Datos". Este proyecto pertenece a la Universidad Nacional de San Luis y se encuentra dentro del Programa de Incentivos a la Investigación (Código 22/F014). El ámbito de este proyecto ha permitido el estudio y tratamiento de objetos de diversos tipos, útiles en distintos campos de aplicación: sistemas de información geográfica, robótica, visión artificial, computación móvil, diseño asistido por computadora, motores de búsqueda en internet, computación gráfica, entre otras, y que se relacionan en tales bases de datos. Se consideran como actividades centrales de esta línea el análisis de distintos tipos de bases de datos, la investigación de aspectos empíricos, teóricos y aplicativos derivados de la administración de una base de datos que maneja tipos de datos no convencionales, la expresividad de los lenguajes de consulta, los operadores necesarios para responder consultas de interés, y también las estructuras y operaciones necesarias para resolverlas eficientemente.

La participación de nuestros integrantes en actividades de cooperación internacional con: Universidad de Chile, Universidad de Massey (Nueva Zelanda), Universidad de Talca (Chile) y el Centro de Investigación Científica y de Educación Superior de Ensenada (México), permite nuevas perspectivas en nuestras investigaciones.

Introducción

Así como las bases de datos tradicionales son diseñadas y optimizadas para resolver eficientemente búsquedas exactas sobre sus datos, realizando comparaciones entre elementos simples, en el nuevo tipo de base de datos que surge como consecuencia de aplicaciones no tradicionales que utilizan datos no estructurados, no siempre es posible definir una clave de búsqueda significativa para cada elemento de la base de datos. En muchos casos hay que usar todo el elemento como clave de búsqueda, lo que

requiere muchas operaciones aritméticas y/o de I/O para procesar una consulta. Además, algunas aplicaciones pueden requerir la búsqueda de elementos similares a un objeto de consulta, aunque la propia consulta no pertenezca a la base de datos.

Todas estas aplicaciones tienen características comunes, englobadas en el modelo de *espacio métrico*. Formalmente, un espacio métrico consiste de un universo de objetos $\mathbb{U}$ y una función de distancia definida entre ellos $d : \mathbb{U} \times \mathbb{U} \longmapsto \mathbb{R}^+$ que mide la (di)similitud entre los objetos. En este ámbito las búsquedas exactas carecen de sentido y es importante la elección de este modelo por las *búsquedas por similitud*, más naturales sobre estos tipos de datos.

El desempeño de los índices existentes, se deteriora exponencialmente con la dimensión del espacio, tanto en espacios de vectores como en espacios métricos, debido a la conocida "*maldición de la dimensionalidad*". Sin embargo no está completamente analizado su efecto sobre los MAMs (*métodos de acceso métricos*). Pocas de las estructuras que existen para búsquedas por similitud en espacios métricos, son eficientes en espacios de alta o mediana dimensión, y la mayoría no admiten dinamismo, ni están diseñadas para conjuntos masivos de datos (en memoria secundaria), lo que permite el estudio de distintas maneras de optimizarlas.

Además del dinamismo en las estructuras y operaciones de búsqueda complejas, se está investigando la obtención de mayor expresividad en los lenguajes utilizados para expresar consultas y caracterizar la clase de consultas computables. Por otro lado, el trabajo con bases de datos masivas, o con aquellas que almacenan objetos muy grandes, da lugar a líneas de investigación que, conscientes del cambio del modelo de costo a considerar, diseñan estructuras de datos más eficientes para memorias jerárquicas.

Líneas de Investigación y Desarrollo

Bases de Datos Métricas

Los espacios métricos serán tomados como modelo para las bases de datos no convencionales. En espacios métricos generales, la complejidad usualmente se mide como el número de cálculos de distancias realizados. Además, al momento de realizar una consulta por similitud eficientemente, es necesario el uso de MAMs, por ello se analizan aquellos que han mostrado buen desempeño en las búsquedas, para optimizarlos más, considerando la jerarquía de memorias. En general, dada una base de datos $X \subseteq \mathbb{U}$ y una consulta $q \in \mathbb{U}$ las consultas son de dos tipos: por *rango* o de k*-vecinos más cercanos*.

Métodos de Acceso Métricos

A partir del *Árbol de Aproximación Espacial* [12], un índice que mostró un muy buen desempeño en espacios de mediana a alta dimensión, pero totalmente estático, se desarrolló uno de los pocos índices completamente dinámicos: el *Árbol de Aproximación Espacial Dinámico (DSAT)* [13] que permite realizar inserciones y eliminaciones, conservando su buen desempeño en las búsquedas. El *DSAT* particiona el espacio considerando la proximidad espacial; pero, si el árbol agrupara los elementos muy cercanos entre sí, lograría mejorar las búsquedas, al evitar recorrerlo. Podemos pensar entonces que construimos un *DSAT*, en el que cada nodo representa un grupo de elementos cercanos ("clusters") y los relacionamos por su proximidad en el espacio. Cada nodo mantiene el centro del cluster correspondiente, y almacena los k elementos más cercanos a él; cualquier elemento a mayor distancia del centro que los k almacenados, forma parte de otro nodo en el árbol [2]. Nuevas estrategias de optimización de funciones a través de heurísticas bioinspiradas, que han mostrado ser útiles en detección de clusters, pueden servir para analizar cuán bueno es el agrupamiento o "clustering"que logra esta estructura.

Dado que una base de datos métrica, ya sea por ser masiva o porque sus objetos son muy grandes, o porque el índice no quepa en memoria principal, o ambas cosas, no se almacene en memoria principal, surge la necesidad de hacer uso de la memoria secundaria. Esto requiere diseñar índices especialmente para memoria secundaria. Así, en [14] se presentaron versiones preliminares del *DSAT* (*DSAT+* y *DSAT**) y se están analizando variantes que mejoren aún más su desempeño. También, se han logrado operaciones eficientes de inserción y eliminación de elementos en las versiones de memoria secundaria del *DSAT*, porque numerosas aplicaciones necesitan del total dinamismo de las estructuras. Además, se está diseñando un nuevo índice dinámico para memoria secundaria, basado en la *Lista de Clusters* [3], que mantenga su buen desempeño en espacios de alta dimensión, que sea dinámico, con buena ocupación de página y eficiente en el número de cálculos de distancia y en operaciones de I/O.

Búsqueda aproximada de los *All-*k*-NN*

En aplicaciones, tales como la clasificación y aprendizaje automático, donde un nuevo elemento debe ser clasificado de acuerdo a sus vecinos más

cercanos, la cuantificación y compresión de imágenes, donde sólo algunos vectores pueden ser representados y los que no deben ser codificados como su punto representable más cercano, la predicción de funciones, en la que desea buscar el comportamiento más similar de una función en el pasado para predecir su comportamiento futuro probable, etc., existen características comunes que exigen estructuras de datos especializadas que las incluyan, como las de *espacios métricos*.

Dado que, como se expresó anteriormente, la evaluación de la función de distancia d suele ser muy costosa, se usa como medida de complejidad en la mayoría de los casos. Aquí existen varias técnicas para resolver el problema de consultas por similitud en un número sublineal de cálculos de distancia, con la condición del *preprocesamiento* de los datos.

Uno de los primitivos básicos de las búsquedas por similitud es la recuperación de los $k-$*vecinos más cercanos*. El mismo puede definirse como: Sea X un conjunto de elementos y d la función de distancia definida entre ellos, los k-NN(u) son los k elementos en $X - \{u\}$ que tengan la menor distancia a u de acuerdo con la función d. Una variante de este problema, quizá menos estudiada, es la búsqueda de los $k-$vecinos más cercanos de *todos los elementos* de X, *All-*k-NN, es decir: Sea $|X| = n$, calcular los k-NN(u_i) para *todos* los u_i en X, por supuesto realizando menos de n^2 cálculos de distancia. En el marco de una etapa de investigación previa, se propusieron y desarrollaron soluciones a este problema, en espacios métricos generales [16, 15], basadas en la construcción del *Grafo de los* k*-vecinos más cercanos* (kNNG). Éste indexa un espacio métrico, requiriendo una cantidad moderada de memoria, y luego se utiliza en la resolución de las consultas por similitud. El kNNG es un grafo dirigido ponderado que conecta cada elemento del espacio métrico mediante un conjunto de arcos cuyos pesos se calculan de acuerdo a la métrica del espacio en cuestión. El desempeño en las búsquedas por similitud de esta propuesta es superior al obtenido utilizando las técnicas clásicas basadas en pivotes.

Por otro lado, el compromiso de tratar de realizar la menor cantidad de cálculos de distancias posibles durante una búsqueda, ha llevado a investigar un enfoque *aproximado* eficiente para resolver estas consultas por similitud. Este enfoque consiste en permitir una relajación en la precisión de la respuesta a fin de obtener una aceleración en la complejidad de la de consulta [18, 3, 20]. El objetivo de la *búsqueda por similitud aproximada* es reducir significativamente los tiempos de búsqueda al permitir algunos errores en el resultado de la consulta. Además de la consulta se especifica un parámetro de precisión ε para controlar cuán lejos queremos el resultado de la consulta del resultado correcto. Un comportamiento razonable para este tipo de algoritmos es acercarse asintóticamente a la respuesta correcta como ε se acerca a cero. Por lo tanto, el éxito de una técnica de aproximación se basa en la resolución del compromiso calidad/tiempo [4]. Esta alternativa a la búsqueda por similitud "exacta"abarca algoritmos aproximados y probabilísticos.

Join Métricos

A pesar de que el modelo de espacios métricos permite cubrir muchos problemas de búsqueda por similitud, en general deja fuera de consideración al operador de ensamble o "join" por similitud, otra primitiva importante [6]. De hecho, a pesar de la atención que esta primitiva ha recibido en las bases de datos tradicionales y aún en las multidimensionales, no ha habido grandes avances para espacios métricos generales. Nos hemos planteado resolver algunas variantes del problema de join por similitud: (1) *join por rango:* dadas dos bases de datos de un espacio métrico y un radio r, encontrar todos los pares de objetos (uno desde cada base de datos) a distancia a lo sumo r, (2) k*–pares más cercanos:* encontrar los k pares de objetos más cercanos entre sí (uno desde cada base de datos).

Para resolver eficientemente estas operaciones hemos diseñado un nuevo índice métrico, llamado *Lista de Clusters Gemelos* (*LTC*) [17], éste se construye sobre ambas bases de datos conjuntamente, en lugar de indexar una o ambas bases de datos independientemente, permitiendo resolver también las consultas por similitud clásicas sobre cada una de las bases de datos independientemente.

Aunque esta estructura ha mostrado ser competitiva y obtener buen desempeño en relación a las alternativas más comunes para resolver el join, aún debe evolucionar mucho para convertirse en una estructura práctica y más eficiente al trabajar con grandes bases de datos métricas. A la fecha se está analizando su combinación con otra clase de índice basada en "permutantes" para resolver el join aproximado de dos bases de datos; para permitir encontrar, rápida y eficientemente, los pares de elementos más similares entre ambas bases de datos, aunque no los obtenga a todos. Así sería posible extender apropiadamente el álgebra relacional como lenguaje de consulta y di-

señar soluciones eficientes para nuevas operaciones, considerando aspectos de memoria secundaria, concurrencia, confiabilidad, etc. Algunos de estos problemas ya poseen solución en bases de datos espaciales, pero no en bases de datos métricas.

Lenguajes de Consulta

La relación existente entre lógica y teoría de bases de datos es muy estrecha y natural, ya que es posible pensar en una base de datos simplemente como una estructura finita, y utilizar las lógicas para expresar consultas sobre éstas. Esto les da una posición central como modelo computacional para el análisis del poder expresivo de los lenguajes de consultas que nos permiten obtener información de una base de datos, siendo relevante como marco teórico para el estudio de las bases de datos.

La mayoría de los lenguajes de consulta sobre bases de datos es equivalente, en su poder expresivo, a FO (First-Order logic). El principal problema es que la expresividad de FO no es lo suficientemente poderosa, porque no alcanza para reflejar ciertas consultas. Esto ha llevado a la búsqueda de una mayor expresividad por medio de diferentes mecanismos de extensión sobre FO utilizados como herramientas de construcción de lógicas más poderosas. Uno de ellos gracias a incorporar cuantificadores que no pueden ser expresados en FO, como *clausura transitiva* y *punto fijo*, entre otros, los que han sido ampliamente estudiados. La idea de agregar cuantificadores es generalizada mediante la noción de *cuantificadores generalizados de Lindström*[8]. Aún así, estas lógicas todavía resultan incompletas, por lo que se analizan lógicas de orden superior, SO (Second-Order Logic), y algunos de sus fragmentos que han demostrado poseer propiedades interesantes sobre las estructuras finitas. Un resultado importante de R. Fagin fue la caracterización del fragmento existencial $SO\exists$[7]. Allí se establece que las propiedades de las estructuras finitas que son definidas por sentencias existenciales de segundo orden coinciden con las propiedades de la clase de complejidad NP, lo cual fue extendido por Stockmeyer [19], estableciendo una relación cercana entre la lógica SO y la jerarquía de tiempo polinomial (PH).

Actualmente existen muchos resultados igualando la expresividad lógica a la complejidad computacional, pero requieren estructuras ordenadas [9, 10]. Estas relaciones entre la complejidad computacional (cantidad de recursos necesarios para resolver un problema sobre algún modelo de máquina computacional) y la complejidad descriptiva (el orden de la lógica que se necesita para describir el problema), han llevado a que los resultados obtenidos en alguno de estos campos sea transferido de manera inmediata al otro.

En uno de nuestros trabajos de investigación se ha introducido la definición de una restricción de SO, que consiste en limitar las relaciones que pueden tomar los cuantificadores de SO, considerando a la lógica como uno de los lenguajes de consulta a base de datos. El tipo de relaciones a los que estos cuantificadores pueden referirse son relaciones cerradas bajo $FO - type$. Esta lógica(SOF) intenta lograr una lógica de mayor poder expresivo que la definida por Dawar(SO^w) en la que los cuantificadores sólo pueden tomar relaciones cerradas bajo $FO - k$ tipos. Se demostró que nuestra lógica incluye estrictamente la de Dawar [5]. Se ha podido definir una nueva clase de complejidad descriptiva (NPF), que caracteriza el fragmento existencial de nuestra lógica gracias a modificar de las máquinas relacionales.

En otro de nuestros trabajos se estudia el impacto del aumento del orden de las variables en las lógicas. Se continúa con el estudio del poder expresivo de las lógicas HO (High-Order logic) y en particular de los fragmentos de la lógica VO (Variable-Order logic) definida en [11], que nace debido a que ninguna de las lógicas de orden superior cubre la clase completa de consultas computables (CQ)[1], es decir que no son completas. Además, si consideramos la unión de todas las lógicas de orden superior, es decir $\bigcup_{i\geq 2} HO^i$ (HO^i representa la lógica de orden i), tampoco obtenemos una lógica completa. De aquí, se define VO permitiendo el uso de variables de orden variable, mediante el uso de cuantificadores de orden. Las restricciones más importantes estudiadas sobre VO son sobre: la cantidad de alternaciones de cuantificadores, la aridad de las variables de orden variable, los valores que pueden asignarse a las variables de orden en función del tamaño del dominio, el rango de cuantificadores y la cantidad de variables, de valuación y de orden.

Resultados y Objetivos

Como trabajo futuro de esta línea de investigación se consideran varios aspectos relacionados al diseño de estructuras de datos que, conscientes de la jerarquía de memorias y de las características particulares de los datos a ser indexados, sean eficientes en espacio y en tiempo.

Por ello, se intentará que los índices se adapten mejor al espacio métrico particular considerado, de-

terminando su dimensión intrínseca, y también al nivel de la jerarquía de memorias donde se almacenará. Estos estudios sobre espacios métricos y sobre algunas estructuras de datos particulares permitirán no sólo mejorar el desempeño de las mismas sino también aplicar, eventualmente, muchos de los resultados que se obtengan a otros MAMs.

Respecto de los lenguajes de consulta se continuará analizando la expresividad de distintas extensiones de FO y posibles restricciones de SO, para lograr caracterizar la clase de las consultas computables sobre bases de datos no convencionales.

Actividades de Formación

Dentro de esta línea de investigación se forman alumnos y docentes-investigadores de acuerdo al siguiente detalle:

Doctorado en Cs. de la Computación: Un integrante de la líinea ha finalizado su tesis sobre la expresividad de la lógica como lenguaje de consulta y otro se encuentra desarrollando su tesis sobre la misma área. Otro integrante desarrolla su tesis sobre bases de datos métricas, esperando su finalización este año.

Maestría en Cs. de la Computación: un investigador de la línea desarrolla su tesis sobre búsqueda por similitud aproximada.

Además, se finalizaron tres trabajos finales de la Licenciatura en Cs. de la Computación, dirigidos por integrantes de la línea.

Referencias

[1] D. Harel A. K. Chandra. Computable queries for relational data bases. *Journal of Computer ans System Sciences*, 21(2):156–178, 1980.

[2] M. Barroso, N. Reyes, and R. Paredes. Enlarging nodes to improve dynamic spatial approximation trees. In *Procs. of the 3rd International Conf. on Similarity Search and Applications (SISAP)*, pages 41–48. ACM Press, 2010.

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

[4] P. Ciaccia and M. Patella. Approximate and probabilistic methods. *SIGSPATIAL Special*, 2(2):16–19, 2010.

[5] A. Dawar. A restricted second order logic for finite structures. *Information and Computation*, 143:154–174, 1998.

[6] V. Dohnal, C. Gennaro, P. Savino, and P. Zezula. Similarity join in metric spaces. In *Proc. 25th European Conf. on IR Research*, LNCS 2633, pages 452–467, 2003.

[7] R. Fagin. Generalized first-order spectra and polynomial-time recognizable sets. *Complexity of Computation*, 7:43–73, 1974.

[8] J. Flum. H. Ebbinghaus. Finite model theory, second edition. *Springer*, 1999.

[9] N. Immerman. Descriptive and computational complexity. *Computational Complexity Theory*, 38:75–91, 1989.

[10] N. Immerman. Descriptive complexity. *Springer*, 1998.

[11] J. M. Turull Torres L. Hella. Computing queries with higher-order logics. *Theoretical Computer Science*, 355:197–214, 2006.

[12] G. Navarro. Searching in metric spaces by spatial approximation. *The Very Large Databases Journal (VLDBJ)*, 11(1):28–46, 2002.

[13] G. Navarro and N. Reyes. Dynamic spatial approximation trees. *Journal of Experimental Algorithmics*, 12:1–68, 2008.

[14] G. Navarro and N. Reyes. Dynamic spatial approximation trees for massive data. In Tomás Skopal and Pavel Zezula, editors, *SISAP*, pages 81–88. IEEE Computer Society, 2009.

[15] R. Paredes. *Graphs for Metric Space Searching*. PhD thesis, University of Chile, 2008.

[16] R. Paredes, E. Chávez, K. Figueroa, and G. Navarro. Practical construction of k-nearest neighbor graphs in metric spaces. In *Proc. 5th Workshop on Efficient and Experimental Algorithms*, LNCS 4007, pages 85–97, 2006.

[17] R. Paredes and N. Reyes. Solving similarity joins and range queries in metric spaces with the list of twin clusters. *J. of Discrete Algorithms*, 7(1):18–35, 2009.

[18] H. Samet. *Foundations of Multidimensional and Metric Data Structures*. Morgan Kaufmann Publishers Inc., CA, USA, 2006.

[19] L. Stockmeyer. The polynomial-time hierarchy. *Theoret. Comput. Sci.*, 3:1–22, 1976.

[20] P. Zezula, G. Amato, V. Dohnal, and M. Batko. *Similarity Search: The Metric Space Approach (Advances in Database Systems)*. Springer-Verlag New York, Inc., USA, 2005. ISBN: 0-387-29146-6.