# Algoritmos Eficientes y Datos Masivos en Búsquedas a Gran Escala

Gabriel H. Tolosa[1,2], Esteban Feuerstein[2]

tolosoft@unlu.edu.ar; efeuerst@dc.uba.ar

[1] Departamento de Ciencias Básicas
Universidad Nacional de Luján
Cruce rutas 5 y 7, Luján.

[2] Departamento de Computación, FCEyN
Universidad de Buenos Aires
Pabellón I, Ciudad Universitaria, Buenos Aires.

## Resumen

La cantidad, diversidad y dinamismo de la información distribuida por diferentes servicios en Internet presenta múltiples desafíos a los sistemas de búsquedas. Por un lado, los usuarios requieren de herramientas que les ayuden a resolver problemas en tiempo y forma. Por otro, el escenario cada vez más grande y complejo y exige el diseño de algoritmos y estructuras de datos que permitan mantener (y mejorar) la eficiencia, tanto en calidad de las respuestas como en tiempo.

Si bien las búsquedas sobre conjuntos masivos de información pueden adquirir formas diversas, una de las aplicaciones más utilizadas son los motores de búsqueda. Éstos son sistemas distribuidos de altas prestaciones que se basan en estructuras de datos y algoritmos altamente eficientes. Esta problemática tiene aún muchas preguntas abiertas y – mientras se intentan resolver cuestiones – aparecen nuevos desafíos.

En este proyecto se propone el diseño y evaluación de estructuras de datos y algoritmos eficientes junto con análisis de datos masivos (big data) para mejoras procesos internos de un motor de búsqueda. Para ello, exploran y explotan tanto el contenido y la estructura de la web como el comportamiento de los usuarios.

**Palabras clave:** motores de búsqueda, estructuras de datos, algoritmos eficientes, datos masivos, big data.

## Contexto

Esta presentación se encuentra enmarcada en el proyecto de tesis de doctorado del primer autor en el Departamento de Computación de la FCEyN (UBA). De forma complementaria, se ha establecido un proyecto de investigación en la UNLu como continuación del proyecto "*Modelos y algoritmos de búsqueda + redes sociales para aplicaciones verticales de recuperación de información*", Departamento de Ciencias Básicas, UNLu.

## Introducción

La cantidad, diversidad y dinamismo en la información disponible a través de diferentes servicios en Internet es cada día más compleja. La red se ha convertido en la más grande y diversa plataforma de comunicaciones que existe[1], en la cual millones de usuarios acceden a diferentes servicios distribuidos de naturaleza diversa y con objetivos particulares [Berners-Lee, 2000] [Wu, 2002] [Baeza-Yates, 2011].

Encontrar información relevante en la web es un desafío, ya que no solamente su dinamismo exige actualización periódica sino que – además – los datos son cada vez mas ricos, complejos y se utilizan y varían en tiempo real. En este escenario, se vuelve un requerimiento que se encuentren disponibles en tiempo y forma [Hall, 2009]. Esto exige que se investiguen y desarrollen nuevas ideas, modelos y herramientas computacionales que permitan satisfacer mas eficientemente las necesidades de acceso, tanto desde la perspectiva de tiempo y espacio como

1 http://www.internetworldstats.com/

de precisión en los resultados [Escudeiro, 2008]. De aquí que los motores de búsqueda se han convertido en herramientas indispensables en la Internet actual.

La arquitectura interna de una máquina de búsqueda de gran escala presenta un grado de complejidad desafiante pero – además – múltiples oportunidades de optimización. Como operan sobre un sistema dinámico y en constante evolución, las soluciones existentes pueden ya no ser eficientes a futuro y nuevas necesidades aparecen constantemente. Sin embargo, dos requerimientos son indispensables: eficiencia (responder en una fracción de segundo a millones de usuarios) y efectividad (que las respuestas sean relevantes).

Paralelamente, la proliferación de grandes volúmenes de datos en casi todos los ámbitos de la actividad humana ha creado una gran demanda de nuevas y poderosas herramientas para convertir datos en información útil. Surgieron así diferentes aportes desde el área de *machine learning* como patrones de reconocimiento, análisis estadístico de datos, visualización, agrupamientos, redes neuronales, entre otros. De igual manera, la disciplina conocida como "Explotación de Datos y Descubrimiento de Conocimiento" (o *Data mining and Knowledge Discovery*) aporta soluciones en múltiples campos. Estos conceptos aplicados al ámbito de la web se lo conoce como Minería Web (*Web Mining*) [Liu, 2008], e incluye el estudio de los datos (minería de contenido), el grafo web (minería de la estructura) y el comportamiento de los usuarios (minería de uso).

En algunos ámbitos, algunas de estas aplicaciones son llamadas también análisis de *Big Data* ya que en sus procesos ingestan grandes volúmenes de datos que requieren ser procesado en poco tiempo [Rajaraman, 2011]. En general, ayudan a resolver problemas que requieren soluciones más complejas y que involucran cómputo paralelo, almacenamiento distribuido y necesitan arquitecturas que puedan escalar de manera flexible [Schadt, 2010]. Como las técnicas para descubrimiento de conocimiento son transversales a cualquier disciplina científica, existe un amplio abanico de soluciones de optimización aún no exploradas para el ámbito de los motores de búsqueda a gran escala que pueden ser tratadas siguiendo la metodología y las técnicas propias de la minería de datos.

Finalmente, el diseño de nuevos algoritmos y estructuras de datos eficientes para manejar datos a gran escala, junto con la integración de la información proveniente de procesos de Big Data pueden permitir incorporar algún grado de "inteligencia"[2] a los sistemas de búsqueda. El estudio de los resultados de estos procesos humanos (y no meramente tecnológicos), como el análisis del comportamiento de los usuarios posibilita también mejorar otros servicios como las búsquedas web y aplicar en nuevos escenarios.

## Líneas de Investigación, Desarrollo e Innovación

En este proyecto, que continúan líneas de I+D iniciadas por el grupo, se propone la incorporación de técnicas de análisis de datos masivos para mejorar los procesos de un motor de búsqueda de gran escala, en ámbitos donde no se ha explorado aún. En algunos casos, se propone el replanteo y/o rediseño de parte de su arquitectura y sus algoritmos internos. En particular, las líneas de I+D principales son:

**1) Estructuras de Datos Distribuidas:** La estructura de datos comúnmente utilizada en un sistema de recuperación de información es el índice invertido. De forma simple, está compuesta por un vocabulario (V) con todos los términos extraídos de los documentos y – por cada uno de éstos – un lista de los documentos donde aparece dicho término junto con información de frecuencia (*posting list*). Para la distribución de la información entre los nodos de búsqueda existen dos enfoques clásicos [Badue, 2001] [Baeza-Yates, 2011]:

a) Particionado por documentos: El conjunto de documentos (C) es dividido entre los P procesadores del sistema, los cuales almacenan una porción del índice C/P. En esta estrategia todos los nodos participan de la resolución de la consulta.

2 No en el sentido estricto del concepto, sino como la idea de "tomar mejores decisiones".

b) Particionado por términos: Cada nodo mantiene información de las listas de posting completas de solamente un subconjunto de los términos. De la forma más trivial, el vocabulario V es dividido entre los P nodos y a cada uno de éstos se le asignan V/P listas. Para la resolución de la consulta solo participan aquellos nodos que poseen la información de los términos involucrados.

También se han propuesto esquemas híbridos como en índice 2D [Feuerstein, 2009] y 3D [Feuerstein, 2012]. La primera consiste en organizar el conjunto de P procesadores en un array bidimensional (C columnas x R filas) en el cual se aplica el particionado por documentos en cada columna y el particionado por términos a nivel de filas. Los resultados de esta estrategia muestran que se pueden obtener mejoras si se selecciona adecuadamente el número de filas y columnas del array. Esto se debe a que existe un *trade-off* entre los costos de comunicación y procesamiento que se requieren para resolver un conjunto de consultas. Para el caso del índice 3D, se agrega una dimensión (D) de procesadores que trabajan como réplicas.

No obstante la estrategia de distribución del índice utilizada, se requieren de la combinación de técnicas y algoritmos para poder responder eficientemente. Además, sobre estas arquitecturas se pueden combinar conjuntos de procesadores para compartir datos.

**2) Algoritmos Eficientes para Búsquedas:** Las técnicas de *caching* se basan en la idea fundamental de almacenar en una memoria de rápido acceso los ítems (objetos) que van a volver a aparecer en un futuro cercano, de manera de poder obtenerlos desde ésta sin incurrir en costos (procesamiento, acceso a disco, entre otros). La idea es explotar la localidad temporal que existe entre pedidos sucesivos de un mismo elemento tratando de mantener en memoria aquellos ítems con mayor chance de que ocurran nuevamente [Podlipnig, 2003].

En motores de búsqueda, habitualmente se implementan caches para resultados de búsqueda [Ozcan, 2008], listas de posting [Zhang, 2008], intersecciones [Long, 2005] y documentos [Strohman, 2007]. Si bien se han propuesto diversos enfoques para cada caso, ninguno está completamente resuelto y aún existen oportunidades de optimización [Marín, 2010]. Por ejemplo, en las caches de resultados, el desafío es poder determinar si una consulta que aparece por primera vez tiene suficientes características como para alojarse en el caché. Dado que la eficiencia de este caché se encuentra acotada por la proporción de consultas únicas (es decir, que solo aparecen una vez) resulta importante maximizarla [Baeza-Yates, 2007]. Inclusive, la frescura de la lista de resultados en caché es un tema abierto dada la dinámica de la colección [Blanco, 2010] [Alici, 2011]. Por otro lado, la optimización de las caches de listas de posting e intersecciones [Feuerstein, 2013] [Feuerstein, 2014] se encuentran relacionadas con el esquema utilizado en la distribución de los documentos.

**3) Big Data en Motores de Búsqueda:** Como se ha mencionado, el análisis de datos masivos aporta múltiples modelos para convertir datos en información útil. Sus técnicas se han utilizado extensamente en motores de búsqueda, por ejemplo: en el análisis del contenido de las páginas, para rankear o personalizar resultados [Mehtaa, 2012]; de la estructura, para optimizar la recolección de páginas (*crawling*) y en el uso, para recomendar consultas similares (q*uery recommendation*) [Anagnostopoulos, 2010].

Sin embargo, no se han utilizado extensivamente para optimizar procesos internos de un buscador por lo que se considera que existen oportunidades de optimización que abren nuevos problemas y temas de investigación. En este proyecto, se pretende utilizar la información proveniente de procesos de descubrimiento para mejorar – entre otros – los casos expuestos previamente: las estructuras de datos distribuidas y los diferentes niveles de memoria caché (por ejemplo, diseñando nuevas políticas de admisión).

## Resultados y Objetivos

El objetivo principal del proyecto es estudiar, desarrollar, aplicar, validar y transferir modelos, algoritmos y técnicas que permitan construir herramientas y/o arquitecturas para abordar algunas de las problemáticas relacionadas con las

búsquedas en Internet. Se pretende estudiar los problemas mencionados relacionados con técnicas de optimización para aplicaciones de búsqueda y proponer mejoras que aumenten la eficiencia de un sistema. Se propone profundizar sobre el estado del arte y definir, analizar y evaluar nuevos enfoques incorporando el análisis de datos masivos a los procesos internos de los motores de búsqueda. En particular:

a) Diseñar estructuras de datos eficientes, en especial aquellas propuestas recientemente a los efectos de evaluar posibles mejoras como las anteriormente descriptas.

b) Determinar, mediante procesos de minería, relaciones entre los objetos del sistema (documento y consultas) y los usuarios externos que permitan establecer mecanismos de resolución de las consultas que aporten mejoras de eficacia (mayor precisión) en la obtención de los resultados.

c) Analizar e implementar técnicas de *caching*, enfocando problema no solamente en las políticas de reemplazo, sino también en políticas de admisión, tema que no ha tenido suficiente desarrollo aún.

d) Diseñar arquitecturas para aplicaciones específicas de búsquedas *ad-hoc* para problemas concretos, donde una solución de propósito general no es la más eficiente. Aquí se deben estudiar cómo los diferentes modelos de distribución de documentos e indexación determinan la eficiencia del sistema

**Dominios de aplicación**

Como se mencionó, la "búsqueda" se convirtió en un proceso central en múltiples aplicaciones basadas en la Web. Los posibles dominios de aplicación de estas técnicas son diversos y se encuentran principalmente relacionados con aquellas aplicaciones donde ocurren un gran número de usuarios (millones) o un volumen de datos considerable como para que su procesamiento no sea trivial (por ejemplo, el tamaño de la web).

Los resultados de estas propuestas se pueden adaptar y transferir a múltiples dominios, principalmente: a) Motores de búsqueda de propósito general, b) Buscadores verticales, c) Redes Sociales y d) Búsquedas móviles.

## Formación de Recursos Humanos

Con este proyecto se espera brindar un marco adecuado para que los algunos docentes auxiliares y estudiantes lleven a cabo tareas de investigación y se desarrollen en el ámbito académico. Junto con el doctorado del primer autor se espera la finalización de una maestría en "Exploración de Datos y Descubrimiento de Conocimiento", DC, FCEyN, Universidad de Buenos Aires.

Se están dirigiendo en temas relacionados con el proyecto tres trabajos finales correspondientes a la Lic. en Sistemas de Información de la Universidad Nacional de Luján y una tesis de Lic. en Ciencias de la Computación del Depto. de Computación de la FCEyN, UBA. Además, se espera dirigir al menos dos estudiantes más por año y al menos un pasante por año realiza tareas con el grupo en el dentro del proyecto.

# Referencias

[Alici, 2011] S. Alici, I. Altingoivde, R. Ozcan, B. Cambazoglu, Ö. Ulusoy. Timestamp-based cache invalidation for search engines. Proc. of the 20th International Conference Companion on World Wide Web. 2011.

[Anagnostopoulos, 2010] A. Anagnostopoulos, L. Becchetti, C. Castillo, and A. Gionis. An Optimization Framework for Query Recommendation. Proc. of the 3rd ACM International Conference on Web Search and Data Mining (WSDM), 2010.

[Badue, 2001] C. Badue, R. Baeza-yates, B. Ribeiro-Neto, N. Ziviani. Distributed query processing using partitioned inverted files. SPIRE Proc. of the 9th String Processing and Information Retrieval Symposium. 2001.

[Baeza-Yates, 2007] R. Baeza-Yates, A. Tiberi. Extracting semantic relations from query logs, in: Proc. of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 76–85. 2007.

[Baeza-Yates, 2011] R. Baeza-Yates and B. Ribeiro-Neto. Modern Information Retrieval: The Concepts and Technology behind Search (2nd Ed). Addison-Wesley Professional. 2011.

[Berners-Lee, 2000] T. Berners-Lee, M. Fischetti, M.L. Dertouzos. Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web. HarperCollins Pub. 2000.

[Blanco, 2010] R. Blanco, E. Bortnikov, F. Junqueira, R. Lempel, L. Telloli, H. Zaragoza. Caching Search Engine Results over Incremental Indices. SIGIR 2010.

[Escudeiro, 2008] N. F. Escudeiro, A. M Jorge. Satisfying Information Needs on the Web: a Survey of Web Information Retrieval. Polytechnical Studies Review, Vol 6, No. 2008.

[Feuerstein, 2009] E. Feuerstein, M. Marín, M. Mizrahi, V. Gil Costa y R. A. Baeza-Yates. Two-dimensional distributed inverted files. In SPIRE 2009, LNCS 5721.

[Feuerstein, 2012] Feuerstein, E; Gil-Costa, V.; Marin, M.; Tolosa G. y Baeza-Yates, R. 3D Inverted Index with Cache Sharing for Web Search Engines, In 18th International European Conference on Parallel and Distributed Computing (Euro-Par 2012), Greece, 2012.

[Feuerstein, 2013] Feuerstein, E. y Tolosa G. Analysis of Cost-Aware Policies for Intersection Caching in Search Nodes. In SCCC Conference, Temucho, Chile, Noviembre de 2013

[Feuerstein, 2014] Feuerstein, E. y Tolosa G. Cost-aware Intersection Caching and Processing Strategies for In- memory Inverted Indexes. In 11th International Workshop on Large-Scale and Distributed Systems for Information Retrieval at WSDM'2014. February, 28. NYC, USA. 2014.

[Hall, 2009] W. Hall, D. De Roure, N. Shadbolt. The evolution of the Web and implications for eResearch. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, Vol. 367, No. 1890, 2009.

[Liu, 2008] Liu, Bing. Web Data Mining: Exploiting Hyperlinks, Contents and Usage Data. Springer, 2008.

[Long, 2005] X. Long, T. Suel. Three-level caching for efficient query processing in large web search engines. In Proc. of the 14th International Conf. on World Wide Web, 2005.

[Marín, 2010] M. Marin, V. Gil-Costa, and C. Gomez-Pantoja. New caching techniques for web search engines. ACM HPDC, 2010.

[Mehtaa, 2012] Pooja Mehtaa, Brinda Parekh, Kirit Modi, and Paresh Solanki. Web Personalization Using Web Mining: Concept and Research Issue. Int. Journal of Information and Education Technology, V2, N5, 2012.

[Ozcan, 2008] R. Ozcan, I. Altingovde, Ö. Ulusoy. Static query result caching revisited. In Proceeding of the 17th International Conference on World Wide Web (pp. 1169–1170). 2008,

[Podlipnig, 2003] S. Podlipnig and L. Boszormenyi. A survey of web cache replacement strategies. ACM Computing Surveys, 35(4):374–398, 2003.

[Rajaraman, 2011] Rajaraman, A., Ullman, J.D., 2011. Mining of massive datasets. Cambridge University Press.. 2011.

[Schadt, 2010] Schadt, E.E., Linderman, M.D., Sorenson, J., Lee, L., Nolan, G.P., 2010. Computational solutions to large-scale data management and analysis. Nature Reviews Genetics 11, 647–657. 2010.

[Strohman, 2007] T. Strohman, W. B. Croft. Efficient document retrieval in main memory. In Proc. of the 30th Annual International ACM SIGIR Conference, 2007.

[Wu, 2002] W. Hu. World Wide Web Search Technologies, Architectural Issues of Web-Enables Electronic Business, Idea Group Publishing, 2002.

[Zhang, 2008] J. Zhang, X. Long, T. Suel. Performance of compressed inverted list caching in search engines. In Proc. of the 17th International WWW Conference. 2008.