Grandes Datos y Algoritmos Eficientes para Aplicaciones de Escala Web

Gabriel H. Tolosa[1], Santiago Banchero[1], Esteban A. Ríssola[1], Tomás Delvechio[1], Pablo Lavallén[1] y Esteban Feuerstein[2]
{tolosoft, sbanchero, earissola, tdelvechio, plavallen}@unlu.edu.ar; efeurest@dc.uba.ar

[1]Departamento de Ciencias Básicas, Universidad Nacional de Luján
[2]Departamento de Computación, FCEyN, Universidad de Buenos Aires

Resumen

La cantidad y variedad de información disponi- ble online impone constantes desafíos en cuanto a técnicas eficientes para su almacenamiento y acceso. Muchos procesos en múltiples dominios requieren que este acceso se realice bajo restricciones de tiempo (eficiencia) y con parámetros de alta calidad (eficacia). En este escenario existen por un lado, necesidades puntuales de los servicios que recolectan y utilizan información de la más diversa y comple- ja naturaleza y por el otro, aparecen oportunidades únicas para avances científico/tecnológicos en áreas como algoritmos, estructuras de datos, sistemas distribuidos y procesamiento de datos a gran escala. Ejemplos concretos son las máquinas de búsqueda para la web, las redes sociales y los sistemas que generan grandes cantidades de datos como la telefonía móvil, entre otros.

Esta problemática abre nuevos interrogantes constantemente y, mientras se intentan resolver, aparecen nuevos desafíos. Algunas de estas preguntas tienen que ver con nuevas estructuras de datos y algoritmos altamente eficientes.

En este proyecto se estudian, proponen, diseñan y evalúan estructuras de datos y algoritmos eficientes junto con el análisis de grandes datos que permitan mejorar las prestaciones de los sistemas, tanto en eficiencia y escalabilidad como en eficacia.

Palabras clave: algoritmos eficientes, motores de búsqueda, estructuras de datos, grandes datos.

Contexto

Esta presentación se encuentra enmarcada en los proyectos de investigación “Algoritmos Eficientes y Minería Web para Recuperación de Información a Gran Escala” del Departamento de Ciencias Básicas (UNLu) y ”Modelos y herramientas algorítmicas avanzadas para redes y datos masivos”del Departamento de Computación de la Facultad de Ciencias Exactas y Naturales (UBA).

Introducción

En las últimas décadas diferentes manifestaciones de los conceptos de redes, datos e información están impactando en la sociedad, tomando diferentes formas y con implicancias aún no conocidas en su totalidad. Aplicaciones en Internet (en particu- lar, en la web), teléfonos con capacidades de cómpu- to considerables, hardware de bajo costo y las redes sociales [12, 11], entre otras conforman un ecosiste- ma en el cual se desarrollan nuevas actividades que ofrecen, además, nuevos desafíos.

En este espacio, dos fenómenos se complementan y retroalimentan. Por un lado, el incremento expo- nencial de la cantidad de datos accesibles a través de las distintas redes y, por el otro, el número de usua- rios y aplicaciones que acceden a éstos. Por esto, existe una necesidad permanente de nuevas ideas algorítmicas y herramientas computacionales que permitan resolver de forma eficiente los problemas que se plantean. Uno de los ejemplos más notables es la web, que ha experimentado en los últimos años un crecimiento en tamaño y complejidad sin precedentes, convirtiéndola en el mayor repositorio de información en el mundo, creando nuevas necesidades de almacenamiento, procesamiento y búsquedas, expandiendo los límites del trabajo en una sola máquina y unos pocos algoritmos al trabajo distribuido, paralelo y altamente eficiente.

En este escenario existen por un lado, necesida-

des puntuales de los servicios que recolectan y utilizan información de la más diversa y compleja naturaleza y por el otro, aparecen oportunidades únicas para avances científico/tecnológicos en áreas como algoritmos, estructuras de datos, sistemas distribuidos y procesamiento de datos a gran escala.

El enfoque más general para acceder a la información en la web es el uso de motores de búsqueda, a partir de consultas basadas en las necesidades de información de los usuarios. De forma simple, los motores de búsqueda intentan satisfacer la consulta de los usuarios realizando procesos de recuperación sobre una porción del espacio web que "conocen", es decir, que han recorrido, recopilado y procesado [4]. Este proceso cuenta con dos características fundamentales: operan con estrictas restricciones de tiem- po, es decir, las consultas deben ser respondidas en pequeñas fracciones de tiempo (milisegundos) y deben ofrecer resultados relevantes a la consulta de los usuarios sobre un escenario altamente heterogéneo. Además, los usuarios no solo *buscan* en la web para satisfacer sus necesidades de información sino que, además, realizan tareas cotidianas (por ejemplo, organizar un viaje, comprar cosas, etc.). Los motores de búsqueda se han convertido en herramientas indispensables y las cuestiones relacionadas con su eficiencia (escalabilidad) y eficacia son temas de muy activa investigación [8].

Esta proliferación de grandes volúmenes de datos y de usuarios en casi todos los ámbitos de la actividad humana ha creado una gran demanda de nue- vas y poderosas herramientas para convertir datos en información útil. Surgieron así diferentes apor- tes desde el área de *machine learning* como patro- nes de reconocimiento, análisis estadístico de datos, visualización, agrupamientos, redes neuronales, entre otros. Estos conceptos y técnicas, aplicados al ámbito de la web se los conoce como Minería Web (Web Mining) [5], e incluye el estudio de los da- tos (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 [35] (Datos Masivos o Grandes Datos) ya que en sus procesos ingestan grandes volúmenes de datos de fuentes diversas [24]. En general, ayudan a resolver problemas que demandan soluciones más complejas y que involucran cómputo paralelo, al- macenamiento distribuido y necesitan arquitecturas que puedan escalar de manera flexible [29], tanto en cómputo como almacenamiento. Como las técnicas para descubrimiento de conocimiento son transver- sales a cualquier disciplina científica, existe un am- plio 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. Incluso, algunas soluciones son significativamente más complejas ya que los volúmenes de información son muy grandes, llegan de manera continua y requieren respuestas en tiempo real [26, 27].

Líneas de investigación y desarrollo

En este proyecto se continúan líneas de I+D del grupo que incorporan análisis de grandes datos en aplicaciones de escala web (como un motor de búsqueda web) que permitan aumentar sus prestaciones. Existen oportunidades de investigación en temas poco explorados por la comunidad científica que permiten mejorar y/o rediseñar los algoritmos internos y las estructuras de datos usadas para recuperación de información de gran escala. En especial, las líneas de I+D principales son:

a. Estructuras de Datos

***1*. Distribuidas:** Los sistemas de búsqueda en texto utilizan como estructura de datos básica un índice invertido, formado por un vocabulario (V) con to- dos los posibles términos y un conjunto de *posting lists* (L) con información acerca de los documentos donde aparece cada término junto con información usada para el ranking. Como los sistemas de búsque- da a gran escala se ejecutan en clusters de compu- tadoras, es necesario distribuir los documentos entre los nodos, ya sea, por documentos [4], por térmi- nos [4] o híbridas (2D [14] y 3D [13]). En todos los casos, el particionado y la asignación de particiones a los nodos de búsqueda impacta en la performance. Resultados experimentales muestran que es posible obtener mejoras si se incorpora la arquitectura del cluster (cantidad de nodos, procesadores y núcleos) en la optimización. Además, los nodos de un motor de búsqueda almacenan su porción del índice en memoria (total o parcialmente), lo que modifica los modelos de costos. Esto ofrece oportunidades para aprovechar de mejor manera el espacio a través del uso eficiente de técnicas de compresión, de reordenamiento y de representación de las listas.

2. Escalables: Para poder mantener la eficiencia conforme se incrementa la cantidad de información que generan algunos servicios (por ejemplo, los si- tios de microblogging) son necesarios algoritmos y estructuras de datos escalables. Los aspectos princi- pales a tener en cuenta en este escenario son la tasa de ingestión de documentos, la disponibilidad inme- diata del contenido y el predominio del factor tem- poral [7, 3]. Para satisfacer estas demandas, resulta indispensable mantener el índice invertido en memo- ria principal. Dado que este es un recurso limitado, se trata de mantener solamente aquella información que permita alcanzar prestaciones de efectividad ra- zonables (o aceptables) [9]. Por ejemplo, el control del crecimiento de las estructuras de datos es un en- foque válido para abordar el problema [27]. Siguien- do esta línea, se propone el desarrollo de una familia de algoritmos de invalidación y poda selectiva [22] de las estructuras de datos a partir del monitoreo online de la evolución y dinámica del vocabulario.

3. Algoritmos Eftcientes: Una de la técnicas más utilizadas para mejorar la performance en motores de búsqueda a gran escala es el *caching*, que se basa en la idea fundamental de almacenar en una me- moria de rápido acceso los ítems que van a volver a aparecer en un futuro cercano. Existen múltiples niveles de caching en una máquina búsqueda, por ejemplo: resultados [23], posting lists [37], intersec- ciones [21] y documentos [31]. Nuestro grupo se en- foca en el problema de las intersecciones para la cual se proponen políticas de admisión y reemplazo que consideren el costo de ejecutar una consulta [15]. Por otro lado, integrar diferentes caches permite optimi- zar el uso de espacio, lo que impacta positivamete en las prestaciones [32].

Una dirección muy interesante es tratar de op- timizar la estrategia de caching incorporando infor- mación proveniente de redes sociales. En trabajos previos del grupo se ha mostrado que los temas que son tendencia en redes sociales guardan relación con el aumento de la popularidad de una consulta rela- cionada al mismo [25] y permiten mejorar la perfor- mance del cache. Esta línea de trabajo es promete- dora ya que el uso de esta clase de información ha mostrado resultados positivos en otros ámbitos (por ejemplo, para mejorar el rendimiento de CDNs).

b. Grandes Datos en Aplicaciones Web

Los motores de búsqueda son probablemente uno de los primeros ejemplo del uso de Grandes Datos. Las demandas de recolección de documen- tos, almacenamiento, analisis, gestión y búsqueda requieren de sofisticados algoritmos que operan so- bre arquitecturas paralelas y distribuidas. Además, la información generada por las búsquedas de los usuarios (consultas, clics, etc.) se convierte en in- formación muy valiosa a partir de la cual es po- sible encontrar patrones de comportamiento y ob- tener estadísticas acerca de cómo los usuarios inter- actúan con los buscadores. Algunos trabajos [16, 18] ya mostraron la potencialidad de estas técnicas.

Esta propuesta global propone optimizar proce- sos internos de un buscador por lo que se conside- ra que existen oportunidades de optimización que abren nuevos problemas y temas de investigación.

c. Plataforma de Procesamiento Distri- buida para Grandes Datos

El los últimos años han aparecido plataformas para procesamiento distribuido en clusters con in- terfaces de alto nivel que *facilitan* el procesamiento distribuido con el costo de montar capas de softwa- re que ofrecen un nivel de abstracción considerable. Los ejemplos clásicos son el sistema de archivos dis- tribuido HDFS [30] y las plataformas Hadoop [34] y Spark [36]. El grupo investiga cómo utilizarlas efi- cientemente en los problemas antes mencionados.

En el caso se las máquinas de búsqueda, un re- querimiento es la indexación distribuida. Los docu- mentos son procesados de forma distribuida y el re- sultado final debe ser un ´ındice invertido particio- nado por algún criterio (como se mencionó anterior- mente) que pueda ser implementado en un cluster. En los últimos años, además, se han propuesto nue- vas estructuras de datos avanzadas que ofrecen un mejor rendimiento en la recuperación (en algunos contextos), como Block-Max [10] y Treaps [19].

Esta línea de investigación se centra en estu- diar, diseñar y evaluar algoritmos de construcción de índices sofisticados como los mencionados utilizan- do estrategias comunmente utilizadas en el ámbito de Grandes Datos (por ej., sobre Hadoop) y tratar de determinar cómo influyen algunos parámetros co- mo el tamaño de la colección y la arquitectura del cluster a utilizar.

Otro ejemplo, es el procesamiento de flujos de *streams*, por ejemplo, datos provenientes de redes sociales o imágenes de cámaras en tiempo real, entre otros. En este caso, se estudia e intenta determinar las condiciones para ejecutar el procesamiento mencionado en clusters utilizando la plataforma Spark para el procesamiento de imágenes mediante algo- ritmos de *machine learning*. La idea es determinar la mejor manera de particionar y distribuir el pro- blema para una arquitectura dada cuyos resultados deben cumplir con restricciones temporales.

d. Algoritmos para Redes Sociales

Las redes sociales online se han convertido sin dudas en una de las aplicaciones más populares de Internet, y han modificado la forma en que los usuarios interactúan e intercambian información. Estas redes atraen a millones de usuarios [6, 17, 20] que, de forma implícita, generan estructuras con propiedades emergentes [1] que surgen del comportamiento global. En general, este tipo de redes tienen a nivel estructural una topología libre de escala (muy sesgada y autosimilar), lo que permite estudiar porciones de la red y extraer propiedades generales. Esto posibilita diseñar algoritmos eficientes para compartir y distribuir la información generada. Esto es especialmente interesante si se tiene en cuenta que la red es un ambiente altamente dinámico y de gran escala. En este caso, si se considera que estos servicios son procesos humanos (y no meramente tecnológicos), su mejor comprensión posibilitará aprovechar la inteligencia colectiva para mejorar servicios como las búsquedas web y aplicar a nuevos escenarios.

e. Comunidades

Otra dirección interesante es en el estudio de algoritmos eficientes para la conformación de comunidades o grupos [33]. Esta es una tarea desafiante a gran escala en aspectos que van desde el tamaño y el tipo de interacción hasta las similitudes por contenido. Si bien existen diversos métodos para analizar y modelar este tipos de redes, la necesidad de algorit- mos que combinen información estructural con las propiedades de los nodos es un requerimiento para un amplio espectro de potenciales aplicaciones con- cretas. Algunos de estos problemas tienen aplicación potencial en proyectos de colaboración abierta, en la salud (grupos de personas con patologias similares) o en el caso de catástrofes [28, 2].

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 a gran escala y el procesamiento de grandes datos. Se pretende estudiar los problemas mencionados relacionados con técnicas de optimización para aplicaciones de búsqueda y proponer mejoras arquitecturales que permitan mejorar la eficiencia de un sis- tema. Se propone profundizar sobre el estado del arte y definir, analizar y evaluar nuevos enfoques incorporando las técnicas de minería de la web a los procesos internos en aplicaciones de escala web. En particular se estudiarán las siguientes líneas principales:

- Estructuras de datos eficientes, en especial aquellas propuestas recientemente a los efec- tos de evaluar posibles mejoras orientadas a problemas de datos masivos.
- Técnicas de caching, enfocando el problema no solamente en las políticas de reemplazo, sino también en políticas de admisión, tema que no ha tenido suficiente desarrollo aún. Aquí se propone un enfoque mediante el uso de técnicas de Web Mining para establecer y aprove- char propiedades de las consultas.
- Arquitecturas para aplicaciones específicas, diseñando aplicaciones de búsqueda ad-hoc para problemas concretos, donde una solución de propósito general no es la más eficiente. Aquí se deben estudiar cómo las estructuras de datos y los algoritmos de búsqueda se complementan de mejor manera para aumentar la eficiencia del sistema.
- Algoritmos para el tratamiento de datos provenientes de redes sociales de interés, interactuando con los motores de búsqueda.

Específicamente,

- Diseñar y evaluar nuevas técnicas que optimicen procesos internos de un motor de búsqueda, utilizando información proveniente de procesos de minería web, aumentando la eficiencia del sistema. En especial, políticas de reemplazo/admisión para diferentes niveles de caché.

- Diseñar y evaluar estructuras de datos ad-hoc (centralizadas y/o distribuidas) para problemas concretos y siguiendo el mismo criterio que en el caso previo (mejorar la eficiencia).
- Determinar, mediante procesos de minería web, relaciones entre los objetos del sistema (documento y consultas) y los usuarios exter- nos que permitan establecer mecanismos de resolución de las consultas que aporten mejoras de eficacia (mayor precisión) en la obtención de los resultados.
- Estudiar las potencialidades de las platafor- mas para procesamiento de datos masivos aplicadas a problemas de búsquedas, princi- palmente para indexación distribuida y opti- mizar su rendimiento a partir de utilizar dife- rentes estrategias y configuraciones.
- Estudiar los flujos de información en redes sociales y su interacción con otros sistemas para el armado automático de comunidades de interés, por ejemplo, grupos de personas con intereses médicos (patologías) afines.

Formación de Recursos Humanos

Este proyecto brinda un marco para que algu- nos docentes auxiliares y estudiantes lleven a cabo tareas de investigación y se desarrollen en el ámbito académico. En el mismo, hay en finalización una te- sis de la maestría en "Exploración de Datos y Descubrimiento de Conocimiento", DC, FCEyN, Universidad de Buenos Aires.

Actualmente, se están dirigiendo cuatro trabajos finales correspondientes a la Lic. en Sistemas de Información de la Universidad Nacional de Luján en temas relacionados con el proyecto. Además, hay dos pasantes alumnos y un becario CIN (Beca de Estímulo a las Vocaciones Científicas). Se espera dirigir al menos dos estudiantes más por año y presentar dos candidatos a becas de investigación.

Referencias

[1] R. Albert and A.-L. Barabási. Statistical mechanics of complex networks. *Rev. Mod. Phys.*, 74, 2002.

[2] A. Anagnostopoulos, L. Becchetti, C. Castillo, *A.* Gionis, and S. Leonardi. Online team formation in social networks. In *Proc. of the 21st International Conference on World Wide Web*, WWW '12, New York, NY, USA, 2012. ACM.

[3] N. Asadi, J. Lin, and M. Busch. Dynamic memory allocation policies for postings in real-time twitter search. *CoRR*, abs/1302.5302, 2013.

[4] R. A. Baeza-Yates and B. A. Ribeiro-Neto. *Modern Information Retrieval - The concepts and technology behind search, 2nd ed.* Pearson Education Ltd., 2011.

[5] L. Bing. *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*. Springer, Secaucus, NJ, USA, 2008.

[6] D. M. Boyd and N. B. Ellison. Social network si- tes: Definition, history, and scholarship. *Journal of Computer-Mediated Communication*, 13(1), 2007.

[7] M. Busch, K. Gade, B. Larson, P. Lok, S. Luckenbill, and J. Lin. Earlybird: Real-time search at twitter. In *Proc. of the 28th International Conference on Da- ta Engineering*, ICDE '12. IEEE Computer Society, 2012.

[8] B. B. Cambazoglu and R. A. Baeza-Yates. Sca- lability and efficiency challenges in large-scale web search engines. In *Proc. of the Eighth ACM Interna- tional Conference on Web Search and Data Mining, WSDM*, 2015.

[9] C. Chen, F. Li, B. C. Ooi, and S. Wu. Ti: An efficient indexing mechanism for real-time search on tweets. In *Proc. of the 2011 ACM SIGMOD International Conference on Management of Data*. ACM, 2011.

[10] S. Ding and T. Suel. Faster top-k document retrie- val using block-max indexes. In *Proc. of the 34th International ACM SIGIR Conferenceon Research and Development in Information Retrieval*, SIGIR '11. ACM, 2011.

[11] P. A. Dreyer Jr. and F. S. Roberts. Irreversible *k* -threshold processes: Graph-theoretical threshold models of the spread of disease and of opinion. *Discrete Applied Mathematics*, 2009.

[12] D. Easley and J. Kleinberg. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge University Press, New York, NY, USA, 2010.

[13] E. Feuerstein, V. G. Costa, M. Mar´ın, G. Tolosa, and R. A. Baeza-Yates. 3d inverted index with cache sharing for web search engines. In *18th International Conference, Euro-Par 2012, August 27-31, 2012.*, 2012.

[14] E. Feuerstein, M. Mar´ın, M. J. Mizrahi, V. G. Costa, and R. A. Baeza-Yates. Two-dimensional distributed inverted files. In *16th International Symposium of String Processing and Information Retrieval, SPIRE'09, August 25-27*, 2009.

[15] E. Feuerstein and G. Tolosa. Cost-aware inter- section caching and processing strategies for in- memory inverted indexes. In *In Proc. of 11th Workshop on Large-scale and Distributed Systems for Information Retrieval*, LSDS-IR'14, 2014.

[16] Y. Hu, Y. Qian, H. Li, D. Jiang, J. Pei, and Q. Zheng. Mining query subtopics from search log data. In *Proc. of the 35th international ACM SIGIR conference on Research and development in informationretrieval*. ACM, 2012.

[17] A. Java, X. Song, T. Finin, and B. Tseng. Why we twitter: Understanding microblogging usage and communities. In *Proc. of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis*, WebKDD/SNA-KDD '07, New York, NY, USA, 2007. ACM.

[18] P. Kaushik, S. Gaur, and M. Singh. Use of query logs for providing cache support to the search engine. In *International Conference on Computing for Sustainable Global Development (INDIACom)*, 2014.

[19] R. Konow, G. Navarro, C. L. Clarke, and A. López-Ort´ız. Faster and smaller inverted indices with treaps. In *Proc. of the 36th International ACM SI- GIR Conference on Research and Development in Information Retrieval*, SIGIR '13. ACM, 2013.

[20] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? In *Proc. of the 19th International Conference on World Wide Web*, WWW '10, 2010.

[21] X. Long and T. Suel. Three-level caching for efficient query processing in large web search engines. In *Proc. ofthe 14th international conferenceon World Wide Web*. ACM, 2005.

[22] A. Ntoulas and J. Cho. Pruning policies for two- tiered inverted index with correctness guarantee. In *Proc. of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2007.

[23] R. Ozcan, I. S. Altingovde, and O. Ulusoy. Cost-aware strategies for query result caching in web search engines. *ACM Trans. Web*, 5(2), May 2011.

[24] A. Rajaraman and J. D. Ullman. *Mining of Massive Datasets*. Cambridge University Press, New York, NY, USA, 2011.

[25] S. Ricci and G. Tolosa. Efecto de los trending topics en el volumen de consultas a motores de búsqueda. In *XVII Congreso Argentino de Ciencias de la Computación, CACIC.*, 2013.

[26] E. R´ıssola and G. Tolosa. Inverted index entry invalidation strategy for real time search. In *Proc. of the XXI Congreso Argentino en Ciencias de la Computación*, CACIC '15, 2015.

[27] E. R´ıssola and G. Tolosa. Improving real time search performance using inverted index entries invalidation strategies. *Journal of Computer Science & Technology*, 16(1), 2016. ISSN: 1666-6038.

[28] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: Real-time event detection by social sensors. In *Proc. of the 19th International Conference on World Wide Web*, WWW '10, New York, NY, USA, 2010. ACM.

[29] E. Schadt et al. Computational solutions to large- scale data management and analysis. *Nature reviews Genetics*, 11(9), 2010.

[30] K. Shvachko, H. Kuang, S. Radia, and R. Chans- ler. The Hadoop distributed file system. In *2010 IEEE 26th Symposium on Mass Storage Systems and Technologies, MSST2010*, 2010.

[31] T. Strohman and W. B. Croft. Efficient document retrieval in main memory. In *SIGIR 2007: Proc. of the 30th Annual International ACM SIGIR Confe- rence on Research and Development in Information Retrieval*, 2007.

[32] G. Tolosa, L. Becchetti, E. Feuerstein, and A. Marchetti-Spaccamela. Performance improvements for search systems using an integrated cache of lists+intersections. In *Proc. of 21st Internatio- nal Symposium of String Processing and Informa- tion Retrieval, SPIRE'14*, 2014.

[33] M. Wang, C. Wang, J. X. Yu, and J. Zhang. Community detection in social networks: An in-depth benchmarking study with a procedure-oriented framework. *Proc. VLDB Endow.*, 8(10), 2015.

[34] T. White. *Hadoop: The Definitive Guide*. O'Reilly Media, Inc., 1st edition, 2009.

[35] W. X. Z. Xingquan, W. Gong-Qing, , and D. Wei. Data mining with big data. *Knowledge and Data Engineering, IEEE Transactions on*, 26(1), 2014.

[36] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shen- ker, and I. Stoica. Spark : Cluster Computing with Working Sets. *HotCloud'10 Proc. of the 2nd USE- NIX conference on Hot topics in cloud computing*, page 10, 2010.

[37] J. Zhang, X. Long, and T. Suel. Performance of compressed inverted list caching in search engines. In *Proc. ofthe 17th international conferenceon World Wide Web*, WWW '08. ACM, 2008.